

**APLIKASI PERHITUNGAN ZAKAT DAN FIDYAH BERBASIS
ANDROID MENGGUNAKAN METODE
*DESIGN THINKING***

SKRIPSI

**AINA MELIA
72154064**



**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SUMATERA UTARA
MEDAN
2019**

**APLIKASI PERHITUNGAN ZAKAT DAN FIDYAH BERBASIS
ANDROID MENGGUNAKAN METODE
*DESIGN THINKING***

SKRIPSI

Diajukan untuk Memenuhi Syarat Mencapai Gelar Sarjana Komputer

**AINA MELIA
72154064**



**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SUMATERA UTARA
MEDAN
2019**

PERSETUJUAN SKRIPSI

Hal : Surat Persetujuan Skripsi

Lamp : -

Kepada Yth.,
Dekan Fakultas Sains dan Teknologi
Universitas Islam Negeri Sumatera Utara Medan

Assalamu'alaikum Wr. Wb.

Setelah membaca, meneliti, memberikan petunjuk, dan mengoreksi serta mengadakan perbaikan, maka kami selaku pembimbing berpendapat bahwa skripsi saudara,

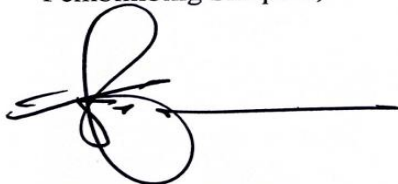
Nama	: Aina Melia
Nomor Induk Mahasiswa	: 72154064
Program Studi	: Sistem Informasi
Judul	: Aplikasi Perhitungan Zakat dan Fidyah Berbasis Android Menggunakan Metode <i>Design Thinking</i>

dapat disetujui untuk segera *dimunaqasyahkan*. Atas perhatiannya kami ucapkan terimakasih.

Medan, 11 November 2019 M
14 Rabiul Awal 1441 H

Komisi Pembimbing,

Pembimbing Skripsi I,



Ilka Zufria, M.Kom.
NIP. 198506042015031006

Pembimbing Skripsi II,



M. Fakhriza, M.Kom.
NIB. 1100000115

SURAT PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertanda tangan di bawah ini,

Nama : Aina Melia
Nomor Induk Mahasiswa : 72154064
Program Studi : Sistem Informasi
Judul : Aplikasi Perhitungan Zakat Dan Fidyah
Berbasis Android Menggunakan Metode
Design Thinking

menyatakan bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya. Apabila dikemudian hari ditemukan plagiat dalam skripsi ini maka saya bersedia menerima sanksi pencabutan gelar akademik yang saya peroleh dan sanksi lainnya sesuai dengan peraturan yang berlaku.

Medan, 25 November 2019



Aina Melia
NIM.72154064



**KEMENTERIAN AGAMA REPUBLIK INDONESIA
UNIVERSITAS ISLAM NEGERISUMATERA UTARA MEDAN
FAKULTAS SAINS DAN TEKNOLOGI**

Jl. IAIN No. 1 Medan 20235
Telp. (061) 6615683-6622925, Fax. (061) 6615683
Url: <http://saintek.uinsu.ac.id>, E-mail: saintek@uinsu.ac.id

PENGESAHAN SKRIPSI

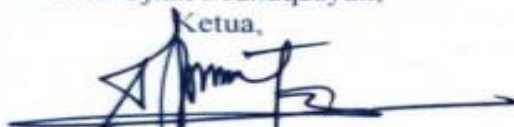
Nomor.061/ST/ST.V/PP.01.1/02/2020

Judul : Aplikasi Perhitungan Zakat Dan Fidyah Berbasis Android
Menggunakan Metode *Design Thinking*
Nama : Aina Melia
Nomor Induk Mahasiswa : 72154064
Program Studi : Sistem Informasi
Fakultas : Sains dan Teknologi

Telah dipertahankan dihadapan Dewan Penguji Skripsi Program Studi Sistem Informasi
Fakultas Sains dan Teknologi UIN Sumatera Utara Medan dan dinyatakan **LULUS**.

Pada hari/tanggal : Kamis, 11 November 2019
Tempat : Ruang Sidang Fakultas Sains dan Teknologi

Tim Ujian Munaqasyah,
Ketua,

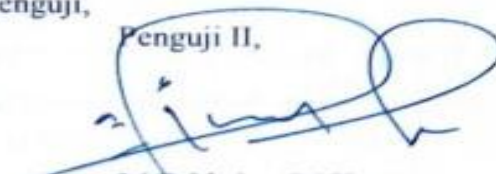

Samsudin, ST., M.Kom
NIP. 197612272011011002

Dewan Penguji,


Penguji I,


Ilka Zulha, M.Kom.
NIP. 198506042015031006


Penguji II,


M.Fakhriza, M.Kom.
NIB. 1100000115

Penguji III,


Samsudin, ST., M.Kom
NIP. 197612272011011002

Penguji IV,


M.Irwan Padli Nasution, ST., M.M., M.Kom
NIP. 1975021320060410003

Mengesahkan,
Dekan Fakultas Sains dan Teknologi
UIN Sumatera Utara Medan,



Dr. H. M. Jamil, M.A.
NIP. 196609101999031002

ABSTRAK

Pada saat ini teknologi informasi mengalami perkembangan yang cukup pesat. Selain menyangkut perangkat keras (*hardware*) dan perangkat lunak (*software*), teknologi juga memperhatikan kepentingan manusia dalam pemanfaatannya. Salah satu kecanggihan teknologi informasi yang digemari oleh manusia adalah android yang telah populer dikalangan *Smartphone*. Singkatnya pada kesempatan kali ini, peneliti melakukan riset dengan tema zakat dan fidyah untuk melihat tingkat pemahaman masyarakat mengenai objek yang diteliti. Sebagaimana telah diketahui bahwa, jika seorang muzakki (pemberi zakat) ingin mengeluarkan zakat, para muzakki mengalami kebingungan mulai dari tatacara perhitungan dan pembagian dari setiap jenis zakat yang berbeda-beda. Hal ini juga terjadi dalam memberikan fidyah puasa yang ditinggalkan. Oleh karena itu dibutuhkanannya pemanfaatan teknologi informasi untuk merancang aplikasi perhitungan zakat dan fidyah yang telah terintegrasi di *mobile application* serta didukung dengan fitur informasi untuk menambah wawasan pengguna. Dengan adanya aplikasi ini tentu mempermudah setiap pengguna. Aplikasi zakat dan fidyah ini dibangun dengan android studio serta menggunakan metode pengembangan sistem yang bernama *Design Thinking*.

Kata Kunci : Aplikasi Android, Fidyah, Zakat, *Design Thinking*.

ABSTRACT

At this time information technology has experienced sufficient development rapidly. Apart from the hardware (hardware) and software (software), technology also pays attention to human interests in utilization. One of the sophistication of information technology favored by humans are androids that have been popular among smartphones. In short On this occasion, researchers conducted research on the theme of zakat and fidyah to see the level of public understanding of the object under study. As it is known that, if a muzakki (giver of zakat) wants issuing zakat, muzakki experience confusion starting from the procedures calculation and distribution of each type of zakat is different. This too occurs in providing the abandoned fasting fidyah. Therefore the need for the use of information technology to design applications zakat and fidyah calculations that have been integrated in the mobile application and on support with information features to increase user insight. With the existence of this application certainly makes it easier for every user. Zakat and fidyah applications This was built with Android Studio and uses the development method a system called Design Thinking.

Keywords: *Android Application, Fidyah, Zakat, Design Thinking.*

KATA PENGANTAR

Assalamualaikum Wr,Wb

Alhamdulillahirobbilalamin segala rasa syukur dan pujian hanya milik Allah SWT atas nikmat, berkah, kasih sayang dan ridhonya sehingga penulis dapat menyelesaikan Proposal Skripsi ini dengan judul “**Aplikasi Perhitungan Zakat Dan Fidyah Berbasis Android Menggunakan Metode *Design Thinking***”. Shalawat dan salam senantiasa dilimpahkan kepada baginda Rasulullah yakni baginda Muhammad SAW beserta sahabat dan para pengikut beliau, dengan mengucapkan Allahumma shalli alaa Muhammad Waala Ali Muhammad, Assalamualaika ya Rasullah

Dalam menyelesaikan skripsi ini, penulis telah banyak mendapatkan bimbingan, dukungan, dan bantuan dari berbagai pihak. Maka dalam kesempatan ini dengan segala kerendahan hati, penulis ingin menyampaikan ucapan terima kasih kepada:

1. Bapak Prof. Dr. Saidurrahman, M. Ag, selaku Rektor Universitas Islam Negeri Sumatera Utara beserta Staf.
2. Bapak Dr. H. M. Jamil, M.A, selaku Dekan Fakultas Sains & Teknologi Universitas Islam Negeri Sumatera Utara.
3. Bapak Samsudin ST.,M.Kom, selaku Ketua Program Studi Sistem Informasi Universitas Islam Negeri Sumatera Utara.
4. Ibu Triase, M.kom, Selaku Seketaris Program Studi Sistem Informasi Universitas Islam Negeri Sumatera Utara.
5. Bapak Ilka Zufria, M.Kom, selaku Dosen Pembimbing 1 yang telah membimbing penulis selama menyelesaikan skripsi ini.
6. Bapak M.Fahkriza, ST, M.Kom, selaku Dosen Pembimbing Akademik dan Pembimbing II yang telah membimbing penulis selama menyelesaikan skripsi ini.

7. Ayahanda Ramli Zein Ahmad, Ibunda Kamelia, kakak saya Rahmalia, SE., adik-adik saya Wulandari, M.akbar, Rabiatul Adawiyah, Ayu Muliani, adik bungsu kecil saya Imam Sholeh dan segenap keluarga beras tercinta. terimakasih banyak atas doa, support, motivasi, materi yang sangat bermakna bagi penulis .
8. Kepada seluruh teman-teman Sistem Informasi angkatan 2015 Fakultas Sains dan Teknologi. Dan seluruh teman-teman lokal Sistem Informasi 2 angkatan 2015 fakultas Fakultas Sains dan Teknologi.
9. Teman dan Sahabat terbaik saya Herawati Hasibuan, Febry Dwi Syafitri, Siti Afzalena, Ella Monica, Rr Balqis Neurita Maharani, Zahra Azura, Suci Humairoh Sitorus, Guslila Sari.

Semoga segala bantuan dan doa yang telah diberikan menjadi amal baik yang akan mendapat balasan dari Allah SWT. Semoga hasil dari Skripsi ini dapat bermanfaat bagi kita semua, amiin.

Medan, 06 November 2019

Penyusun,

Aina Melia

DAFTAR ISI

ABSTRAK	i
ABSTRACT	ii
KATA PENGANTAR.....	iii
DAFTAR ISI	v
DAFTAR GAMBAR	viii
DAFTAR TABEL	x
DAFTAR LAMPIRAN	xi

BAB I PENDAHULUAN

1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	4

BAB II LANDASAN TEORI

2.1 Pengertian Aplikasi	7
2.2 Aplikasi Mobile.....	7
2.3 Android	8
2.3.1 Versi Android.....	10
2.3.2 Fitur-Fitur Android	16
2.3.3 Komponen Aplikasi Android	18
2.4 Sejarah java	19
2.4.1 Java.....	20
2.5 Kotlin	21
2.6 Dart.....	22
2.6.1 Tipe Data Pada Dart	22
2.7 Java SE Development Kit (JDK)	23
2.8 Android SDK Platform	23
2.9 Android Development Tools (ADT).....	23
2.10 Android Virtual Device (ADT)	24
2.11 Android Studio	24
2.12 UML (Unified Modeling Language).....	25
2.12.1 Bangun Dasar UML	26
2.12.2 Tujuan <i>Unified Modelling Language</i> (UML)	33
2.12.3 <i>Tool-Tool</i> yang Mendukung UML.....	34
2.13 HIPO.....	34

2.14 <i>Design Thinking</i>	36
2.14.1 <i>Emphatize</i>	37
2.14.2 <i>Define</i>	38
2.14.3 <i>Ideate</i>	38
2.14.4 <i>Prototype</i>	39
2.14.5 <i>Test</i>	39
2.15 Pengertian Zakat dan Dalil Hukum	39
2.15.1 Harta Yang Wajib di Zakati	40
2.15.2 Jenis Harta Yang Tidak di Zakati	41
2.15.3 Menghitung Zakat Binatang Ternak	43
2.15.4 Menghitung Zakat Profesi atau Penghasilan.....	45
2.15.5 Zakat Fitrah	46
2.15.6 Zakat Fitrah dan Makanan	48
2.15.7 Zakat Emas.....	49
2.15.8 Golongan Orang Yang Menerima Zakat.....	49
2.16 Fidyah Puasa.....	51
2.16.1 Hakikat Puasa	52
2.16.2 Sebab Mengganti Puasa.....	52
2.16.3 Orang Yang Membayar Puasa.....	54

BAB III METODOLOGI PENELITIAN

3.1 Tempat dan Waktu Pelaksanaan	56
3.2 Cara Kerja	59
3.2.1 <i>Emphatize</i>	59
3.2.1.1 Observe (amati).....	59
3.2.1.2 Wawancara.....	60
3.2.1.3 <i>Engage</i> (terlibat).....	60
3.2.1.4 <i>Immerse</i> (resakan langsung).....	62
3.2.1.5 Study Literatur	62
3.2.2 <i>Define</i> (menentukan).....	63
3.2.2.1 Kebutuhan Sistem	63
3.2.3 Kerangka Berpikir.....	65

BAB IV HASIL DAN PEMBAHASAN

4.1 <i>Ideate</i> (ide).....	66
4.1.1 <i>Requirement Planning</i>	66
4.1.1.1 Analisa Sistem Berjalan	66
4.1.1.2 Gambaran Analisis Sistem Berjalan	67
4.1.1.3 Narasi Sistem Berjalan	67
4.1.1.4 Identifikasi Masalah Sistem Berjalan.....	68
4.1.1.5 Analisis Sistem Berjalan	68
4.1.1.6 Gambaran Analisis Sistem Usulan.....	69
4.1.1.7 Narasi Sistem Usulan.....	70
4.1.2 Hipochart	70
4.1.2.1 Penjelasan Hipochart Aplikasi	71

4.2 <i>Prototype</i>	73
4.2.1 <i>Prototype Diagram</i>	73
4.2.1.1 Use Case.....	74
4.2.1.2 <i>Activity Diagram</i>	75
4.3.1.3 <i>Sequance Diagram</i>	79
4.2.2 <i>Prototype Interface</i>	81
4.3 <i>Test</i> (pengujian).....	87
4.3.1 Menentukan Hak Akses	88
4.3.2 <i>Black Box Testing</i>	88
4.3.3 Implentasi	90
 BAB V KESIMPULAN DAN SARAN	
5.1 Kesimpulan	101
5.2 Saran.....	101
 DAFTAR PUSTAKA	102
LAMPIRAN (SOURCE CODE)	

DAFTAR GAMBAR

Gambar	Judul Gambar	Halaman
2.1	Tampilan Android untuk Espler Launcher.....	9
2.2	Aneka Produk Android dari Google.....	9
2.3	Nama OS Android Menurut Versi	10
2.4	Tampilan Android Cupcake, Donut, dan Éclair	11
2.5	Tampilan Android Proyo dan Gingerbread	12
2.6	Tampilan Android Honaycomb	12
2.7	Tampilan Ice Cream Sandwich, Jelly Bean dan Kitakat	13
2.10	Diagram Tahapan Dalam Metode Design Thinking	36
2.11	Tahapan <i>Emphatize</i>	37
2.12	Tahapan <i>Define</i>	38
2.13	Tahapan <i>Ideate</i>	38
3.1	Diagram Flow Metode Design Thinking	59
3.2	Grafik <i>Immerse</i>	62
3.3	Kerangka Berpikir	65
4.1	Sistem Berjalan Pada Mesjid	67
4.2	Sistem Berjalan Pada masyarakat	67
4.3	Analisa Sistem Usulan	69
4.4	Hipochart Aplikasi	71
4.5	<i>Use Case Diagram</i> Aplikasi	74
4.6	Activity Diagram Menu Zakat	75
4.7	Activity Diagram Menu Fidyah.....	76
4.8	Activity Diagram Menu Hitung Zakat	77
4.9	Activity Diagram Menu Hitung Fidyah	78
4.10	<i>Sequance Diagram</i> Menu Zakat	79
4.11	<i>Sequance Diagram</i> Menu Fidyah	80
4.12	<i>Sequance Diagram</i> Menu Hitung Zakat	80
4.13	<i>Sequance Diagram</i> Menu Hitung Fidyah	81
4.14	Dasboard (Menu Utama)	82
4.15	Sub Menu dari Menu Zakat	82
4.16	Sub Menu dari Menu Fidyah	83
4.17	Menu Hitung Zakat Ternak	84
4.18	Menu Hitung Zakat Emas	85
4.19	Menu Hitung Zakat Profesi	86
4.20	Menu Hitung Zakat Fitrah	87
4.21	Tampilan Menu Utama Aplikasi Zakat dan Fidyah	90
4.22	Tampilan Drawel	92
4.23	Sub Menu dari Menu Zakat	93
4.24	Sub Menu dari Menu Fidyah	94
4.25	Tampilan Menu Hitung Fidyah	95
4.26	Tampilan Menu Zakat Ternak	96
4.27	Tampilan Menu Zakat Emas	97

4.28	Tampilan Menu Zakat Fitrah	98
4.29	Tampilan Zakat Profesi	99

DAFTAR TABEL

Tabel	Judul Tabel	Halaman
2.1	Fitur- Fitur Android	17
2.2	Simbol Use Case Diagram	26
2.3	Simbol Sequence Diagram	29
2.4	Simbol Activity Diagram	32
2.5	Pembagian Zakat	42
3.1	Jadwal Penelitian	56
3.2	Data Masyarakat yang Mengisi Kuesioner	60
4.1	Penjelasan Hipochart	71
4.2	<i>Black Box Testing</i>	88
4.3	Penjelasan Menu Utama	89
4.4	Sub Menu dari Menu Zakat	91
4.5	Sub Menu dari Menu Fidyah	93
4.6	Menu Perhitungan Fidyah	94
4.7	Penjelasan Menu Zakat Ternak	95
4.8	Penjelasan Menu Zakat Emas	96
4.9	Penjelasan Menu Zakat Fitrah	97
4.10	Penjelasan Zakat Profesi	98

DAFTAR LAMPIRAN

1. Source Code

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pada saat ini teknologi informasi mengalami perkembangan yang cukup pesat. Teknologi informasi dapat di maksud sebagai kegiatan pengumpulan pengolahan - pengolahan, penyimpanan, penyebaran dan pemanfaatan suatu informasi. Selain perkembangan jenis peralatan teknologi maupun *software* aplikasi pendukung, perkembangan ini juga berdasarkan pada semakin meratanya pengguna teknologi informasi. Selain menyangkut perangkat keras (*hardware*) dan perangkat lunak (*software*), teknologi juga memperhatikan kepentingan manusia dalam pemanfaatannya. Salah satu kecanggihan teknologi informasi yang di gemari oleh manusia adalah android yang telah populer di kalangan *Smartphone*. Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat bergerak layar sentuh seperti telepon pintar dan komputer tablet. Pada kesempatan ini, penulis mencoba memanfaatkan kecanggihan teknologi untuk menjalankan aplikasi perhitungan zakat karena rendahnya pengetahuan masyarakat mengenai zakat yang berakibat tidak efektifnya dalam pengelolaan dan perhitungan jumlah zakat yang akan di keluarkan.

Zakat secara istilah ialah mengeluarkan sebagian dari harta benda atas perintah allah sebagai sedekah wajib kepada mereka yang telah di tetapkan menurut syariat dan hukum agama islam. Zakat itu sebuah karunia karena dengan membayar zakat hartanya akan bertambah atau tidak berkurang sehingga akan menjadikan harta nya tumbuh dengan keberkahan yang diberikan Allah SWT kepada seorang muzakki, dan suci dari kotoran dan dosa yang menyertai yang disebabkan oleh harta yang dimiliki tersebut. Sedangkan fidyah dalam kitab kitab fiqih, di kenal dengan istilah *ith'am* yang artinya memberi

makan. Adapun fidyah yang kebanyakan di bahas di dalam mazhab fiqih ialah, sesuatu yang harus di berikan kepada orang miskin, berupa makanan, sebagai pengganti karena meninggalkan sholat dan puasa.

Dari uraian di atas masyarakat muslim mulai menghadapi beberapa masalah khususnya dalam perhitungan pembagian zakat yang telah sampai satu tahun dan pembayaran fidyah. Masyarakat sering mengalami kebingungan ketika hendak mengeluarkan zakat mulai dari perhitungannya, cara pembagiannya, batas harta yang wajib dikeluarkan zakatnya dan waktu untuk zakat itu di keluarkan, begitu pula asbab dan sebab dari pembayaran fidyah. Dengan adanya permasalahan tersebut teknologi informasi memberikan solusi serta menciptakan inovasi yang baru di era digitalisasi dengan memanfaatkan sejuta kecanggihan teknologi.

Beberapa penelitian telah dilakukan diantaranya di lakukan oleh (Atunnisa, Satria, & Cahyana, 2014), dengan judul “*Pengembangan Aplikasi Zakat Berbasis Android menggunakan Metode Prototype*” dalam penelitiannya, Rifa’atunnisa, dkk mengangkat 3 jenis zakat yang bisa di hitung serta menggunakan metode *prototype* untuk mengembangkan aplikasi mulai dari pengumpulan kebutuhan dan perbaikan, perancangan cepat, membentuk *prototype*, evaluasi pelanggan terhadap *prototype* dan lain-lain. Adapun jenis jenis zakat yang dapat di kalkulasi dalam penelitiannya adalah zakat mall, zakat fitrah dan zakat profesi.

Pada tahun yang sama penelitian ini juga di lakukan oleh (M & M, n.d.) dengan judul “*Rancangan Aplikasi Menghitung zakat Berbasis Android*” dalam penelitian ini aplikasi yang jelaskan tentang perhitungan zakat, yaitu zakat fitrah, zakat maal, zakat profesi, zakat perdagangan, dan zakat pertanian. Adapun tahapan yang akan di lalui dalam penelitian ini adalah perancangan aplikasi, perancangan struktur navigasi dan UML, serta perancangan tampilan untuk mempermudah dalam pembuatan aplikasi.

Dari kedua penelitian yang telah dilakukan sebelumnya, hanya membuat sistem perhitungan zakat yang berbasis android. Maka dengan ini, peneliti mengembangkan beberapa fitur dari penelitian sebelumnya yaitu menambahkan fitur zakat ternak dan pembayaran fidyah guna menciptakan suatu inovasi dalam mendominasi kebutuhan masyarakat untuk memudahkan dalam perhitungan zakat dan pembayaran fidyah berbasis android menggunakan metode *Design Thinking*, metode *design thinking* dikenal sebagai suatu proses berpikir komprehensif yang berkonsentrasi untuk menciptakan suatu solusi, *design thinking* adalah metode penyelesaian masalah yang berfokus pada pengguna atau *user*. *Design thinking* terdiri dari 5 tahapan yaitu *emphitize*, *define*, *ideate*, *prototype* dan *test* . sehingga untuk memfasilitasi tujuan masyarakat dalam berhitung pembagian zakat dan fidyah serta pengetahuan mengenai jenis jenis zakat dan fidyah maka peneliti memberi judul ”Aplikasi Perhitungan Zakat dan Fidyah Berbasis Android Menggunakan Metode *Design Thinking*”. Semoga penelitian ini dapat memecahkan masalah yang terjadi di tengah tengah masyarakat.

1.2 Rumusah Masalah

Berdasarkan uraian dari latar belakang diatas, terdapat beberapa permasalahan yang dapat diidentifikasi, yaitu:

1. Bagaimana merancang sebuah aplikasi untuk menghitung zakat berbasis android ?
2. Bagaimana merancang sebuah aplikasi untuk perhitungan fidyah dan Zakat?
3. Bagaimana menerapkan metode *Design Thinking* dalam perancangan aplikasi perhitungan zakat dan fidyah ?

1.3 Batasan Masalah

Dalam penyusunan penelitian ini peneliti membatasi masalah masalah sebagai berikut:

1. Peneliti membahas tentang pembagian zakat yang akan di keluarkan oleh HGVHVseorang muzakki.
2. Dalam proses perancangan aplikasi pada penelitian ini menggunakan metode *Design Thinking*.
3. Aplikasi perhitungan zakat pada penelitian ini, menampilkan jumlah zakat yang harus dikeluarkan dari hasil zakat profesi, zakat firtah, zakat mal dan zakat ternak.
4. Aplikasi ini bisa menghitung jumlah fidyah yang harus di keluarkan oleh seseorang yang meninggalkan puasa.

1.4 Tujuan Penelitian

Adapun tujuan penelitian ini adalah:

1. Merancang aplikasi yang bisa menghitung zakat dengan efektif , efesien serta terkomputerisasi.
2. Mengembangkan aplikasi perhitungan zakat dengan menambah kontribusi zakat ternak dari penelitian sebelumnya.
3. Melengkapi aplikasi dengan perhitungan fidyah dari penelitian sebelumnya.

1.5 Manfaat Penelitian

Adapun manfaat dari perancangan aplikasi ini adalah:

1. Dengan adanya aplikasi perhitungan zakat ini, pemberi zakat (muzakki) bisa mengetahui jumlah zakat yang harus di keluarkan.
2. Pengguna atau *user* tidak perlu menghitung zakat dan fidyah secara manual karena telah terintegrasi di *application* android.
3. Aplikasi ini dapat di akses oleh pengguna (*user*) atau panitia penyelenggara zakat untuk memberi dan menghitung jumlah zakat yang telah di tetapkan syariat islam.

BAB II

LANDASAN TEORI

2.1 Pengenalan Aplikasi

Aplikasi merupakan program komputer atau perangkat lunak yang dirancang atau didesain untuk mengerjakan tugas tertentu. Sedangkan secara istilah aplikasi adalah suatu program yang siap untuk digunakan yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta pengguna aplikasi lain yang dapat digunakan yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta pengguna aplikasi lain yang dapat digunakan oleh suatu sasaran yang akan dituju hal ini di kemukakan oleh, (Juansyah, 2015).

Menurut kamus komputer eksekutif, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputasi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan.

2.2 Aplikasi *Mobile*

Membicarakan tentang teknologi *mobile* atau aplikasi *mobile* sekarang ini tentu amat menyenangkan. Ini karena perkembangan device *mobile* sangat cepat dan menarik. Salah satunya device android. Perkembangan android yang cepat ini, selain karena kecanggihannya juga karena android merupakan sistem operasi *mobile* yang gratis dan *open source* atau kode sumbernya terbuka. (komputer, Membuat Aplikasi Android untuk Tablet dan Handphone, 2012).

Aplikasi *mobile* adalah aplikasi yang bisa digunakan secara berpindah pindah tempat (*mobile*) sehingga anda bisa menikmati aplikasi kapan pun dan dimana pun pada *smartphone* sambil menjalankan aktifitas rutin anda. Contoh aplikasi *mobile* yang sering digunakan adalah *game*, *web*, *browser*,

sic, *radion*, dan sebagainya. Untuk pengembangan secara mobile, J2ME merupakan (*java 2 micro edition*) adalah program yang sering digunakan pada pager, PDA, dan pada smartphone sampai saat ini. J2ME merupakan program yang menggunakan bahasa pemrograman java yang komponen komponennya terdiri dari *Java Virtual Machine* (JVM) yang digunakan untuk menjalankan program emulator atau *handheld device* dan java API (*Application Programming Interface*) dan *tools* lain yang untuk mengembangkan aplikasi java seperti *Java Emulator*, *Emulator Motorola*, dari J2ME *Wireless Toolkit*, J2ME merupakan satu set spesifikasi dari teknologi yang fokus pada perangkat konsumen, perangkat ini memiliki jumlah memori yang terbatas, menghabiskan sedikit daya baterai, layar kecil dan jaringan yang rendah. (komputer, Membangun Aplikasi Mobile Cross Platform Dengan Phonegap, 2014).

2.3 Android

Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat bergerak layar sentuh seperti telepon pintar dan komputer tablet. Android awalnya dikembangkan oleh android, inc., dengan dukungan finansial dari google, yang kemudian membelinya pada tahun 2005. Sistem operasi ini di liris secara resmi pada tahun 2007, bersamaan dengan didirikannya *Open Handset Alliance*, konsorsium dari perusahaan-perusahaan perangkat keras, perangkat lunak, dan telekomunikasi yang bertujuan untuk memajukan standar terbuka perangkat seluler. Ponsel android pertama mulai di jual pada bulan Oktober 2008.. (yudha yudhanto, 2017).

Android merupakan OS (*Operating System*) *Mobile* yang tumbuh di tengah OS lainnya yang berkembang dewasa ini, hal ini di kemukakan oleh Hermawan (2011). Sedangkan Menurut (wadi, 2018), android merupakan sistem operasi yang berbasis linux dan bersifat kode sumber terbuka (*open source*) untuk peralatan mobile seperti *smartphone* dan tablet. Tidak hanya itu, sistem operasi ini juga digunakan sebagai sistem operasi pada *tvsmart* dan perangkat elektronik cerdas lainnya.



Gambar 2.1 Tampilan Android untuk *Esplero Launcher*
(yudha yudhanto, 2017).

Antarmuka pengguna Android umumnya berupa manipulasi langsung menggunakan gerakan sentuh yang serupa dengan tindakan nyata, misal menggeser, mengetuk, dan mencubit untuk memanipulasi objek di layar, serta papan ketik virtual untuk menulis teks. Selain perangkat layar sentuh, google juga telah mengembangkan Android TV untuk televisi, Android Auto untuk mobil, dan Android Wear untuk jam tangan, Masing masing memiliki antarmuka pengguna yang berbeda. Varian Android juga gunakan pada komputer jingjing, konsol permainan, kamera digital, dan peralatan elektronik lainnya.



Gambar 2.2 Aneka Produk Android dari Google
(yudha yudhanto, 2017).

2.3.1 Versi Android

Sistem operasi Android terdiri dari beberapa versi. Setiap versi Android terbaru memiliki nama unik tersendiri dan memiliki beberapa jenis kelebihan mulai dari tampilan hingga optimasi keamanan. Berikut daftar nama OS Android menurut versi.



Gambar 2.3 nama OS Android menurut versi.

(yudha yudhanto, 2017).

1. Android versi 1.0 diliris tanggal 23.09.2008.
2. Android versi 1.1 diliris tanggal 09.02.2009.
3. Android versi 1.5 *Cupcake* diliris tanggal 30.04.2009.

Merupakan versi pertama yang di luncurkan secara komersil dan juga pertama dinamai berdasarkan nama kue. Android versi ini baru mulai menambahkan fitur-fitur yang sudah lazim di temukan saat ini seperti *Widget*, *Auto Rotate*, dan *keyboard* virtual dari pihak ketiga. Dalam tahap ini bisa di katakan kalau Google masih mencoba untuk memasukkan fitur-fitur dasar.

4. Android versi 1.6 *Donut* diliris tanggal 15.09.2009.

Pada android donut, *smartphone* mendapatkan beberapa fitur tambahan seperti dukungan CDMA, mesin teks ke suara, dan indicator penggunaan baterai. Nomor versi tidak jauh berbeda dibandingkan dengan sebelumnya membuat Android Donut lebih sebagai update mini.

5. Android versi 2.0 *Éclair* di liris tangan 26.10.2009.

Ada perubahan yang cukup banyak di bandingkan versi Android awal. Pada *Update* ini Google memberikan dukungan dan fitur baru yang berfungsi untuk mempermudah para pengguna. Android Éclair mendapatkan beberapa fitur baru yang utamanya pada aplikasi kamera dengan menambahkan dukungan *flash*, *focus*, hingga efek warna .pada versi ini Google juga menambahkan *Live Wallpaper* yang cukup keren pada awal kemunculannya.



Gambar 2.4 Tampilan *Android Cupcake, Donut, dan Éclair*
(yudha yudhanto, 2017).

6. Android versi 2.2 Froyo, diliris 10.05.2010. Pada versi ini android mengalami optimalisasi performa yang cukup signifikan dan di tambah dengan dukungan USB *Tethering*, Wi-Fi Hotspot, dan *push notification* yang sangat berguna hingga saat ini. Selain itu pada versi ini kita bisa memindahkan aplikasi ke penyimpanan eksternal jika kemungkinan.
7. Android versi 2.3 *Gingerbread* tergolong paling sukses, diliris tanggal 06.12.2010.



Gambar 2.5 Tampilan Android Proyo dan *Gingerbread*
(yudha yudhanto, 2017)

8. Android versi 3.0 *Honeycomb* memiliki tampilan mewah dan kinerja paling baik. Diliris tanggal 22.02.2011.

Sistem operasi satu ini di khususkan untuk perangkat *tablet*. Pada *honeycomb* ini beberapa fitur berguna masih bertahan hingga sekarang seperti sistem bar, dukungan prosesor *multi-core*, hingga layar *Home* yang dapat di kustomisasi. Android 3.0 *Honaycomb* pada saat peluncurannya terlihat sangat futuristik dan menarik banyak perhatian.



Gambar 2.6 Tampilan *Android Honeycomb*
(yudha yudhanto, 2017)

9. Android versi 4.0 *Ice Cream Sandwich* sudah mendukung *Flash Player*, di liris tanggal 19,10,2011.

Google mempersatukan perangkat *smartphone* dan *table* serta menambahkan tampilan antarmuka yang lebih minimalis. Salah satu fitur baru yang cukup menarik pada Android versi ini adalah Android Beam yang memungkinkan transfer data secara cepat menggunakan NFC.

10. Android versi *Jelly Bean* memiliki kelebihan di baterai, navigasi *gesture*, dan kamera. Dirilis tanggal 09.07.2012.

Versi ini lebih fokus pada peningkatan kinerja dan keamanan. Sebagian besar peningkatan yang ada versi android ini terjadi pada sistem android itu sendiri. Namun ada satu fitur yang benar benar perlu di ketahui, yakni dukungan resolusi UHD 4K.



Gambar 2.7 Tampilan *Ice Cream Sandwich*, *Jelly Bean*, dan *Kitkat*
(yudha yudhanto, 2017)

11. Android versi 4.4 *Kitkat* diliris tanggal 31.10.2013.

Fokus terhadap peningkatan *user experience* dan performa khususnya pada perangkat kelas bawah. Android 4.4 *KitKat* memberikan batas minimal RAM tersedia untuk Android sebesar 512 MB sehingga perangkat yang memiliki RAM di bawah golongan sebagai perangkat RAM rendah. Fitur terbarunya adalah:

- a. *Screen recording*.
- b. *New Traslucent system UI*.

- c. Peningkatan akses notifikasi.
- d. System-wide setting untuk closed captioning.
- e. Peningkatan kinerja.
- f. Mengaktifkan sprint spark band 26 dan band 41.
- g. Perbaikan lkerentanan pada heartbleed/openSSL.

12. Android versi 5.0 *lollipop* dirilis 17.10.2014.

Merupakan yang pertama kali mengusung desain material yang masih menjadi tren hingga saat ini. Fiturnya adalah:

- a. Desain baru (material)
- b. Peningkatan kecepatan.
- c. Perbaikamn video playback dan kegagalan password.
- d. Support multi SIM cards.
- e. Shortcuts pada quick setting untuk Wi-Fi dan bluetooth.
- f. Pengamanan jika perangkat hilang atau dicuri.
- g. Penggunaan suara high definision untuk telepon.
- h. Peningkatan kinerja stabilitas.

13. Android versi 6.0 *Marshmallow* diliris 28.05.2015.

Semakin memperkuat apaa yang telah dilakukan oleh google selama ini melalui android lollipop. Melalui sistem operasi terbaru ini, google membawa fitur-fitur canggih yang sangat berguna seperti:

- a. Support USB Type-C.
- b. Support autentikasi sidik jari (*fingerprint*)
- c. Daya tahan baterai lebih meningkat dengan manajemen konsumsi baterai oleh doze.
- d. Akses System UI Tuner.
- e. Support sistem pembayaran dengan android pay yang berkolaborasi dengan *fingerprint authentication* sehingga terjamin keamanannya.
- f. Tambahan fungsi google *now* yang tidak sekedar melayani perintah suara.

14. Android versi 7.0 *nougat* diperkenalkan 22.08.2016.

Memberikan peningkatan terhadap pengalaman penggunaan agar lebih mudah. Fitur dukungan *multi-window* menjadi perhatian utama sehingga pengguna dapat menggunakan beberapa aplikasi secara bersama. Selain itu juga ada fitur yang membuat pembaruan sistem operasi lebih mulus dan tidak terasa saat perangkat sedang tidak digunakan.

- a. *Support* *multi-window*.
- b. Langsung balas pesan dari jendela atau menu notifikasi.
- c. Tampilan panel notifikasi dan *quick* settings yang baru.
- d. Mode doze yang ditinggalkan (doze mode 2.0).
- e. Menu di antara system settings.

15. android versi 8.0 oreo dirilis resmi pada tanggal 21.08.2017.

- a. Android O lebih fokus pada kecepatan dan efisiensi.
- b. Kecepatan boot up 2x lebih cepat.
- c. Mode picture lebih *flexible* dari android N.
- d. Aplikasi yang berjalan di latar belakang lebih di perketat untuk menghemat baterai.
- e. Baterai lebih tahan lama.
- f. Emoji yang diperbaharui dan lebih banyak.

Jumlah pengguna android sampai dengan oktober 2017. Urutan terbanyak adalah android marshmallow, lollipop, nougat, dan kitkat. Ketika kita memilih bahwa aplikasi akan digunakan pada marshmallow, maka akan dipastikan pengguna terbanyak tidak akan mendapatkan banyak masalah ketika menjalankan aplikasi yang kita tawarkan.

Selain versi android, yang kita pertimbangkan adalah ukuran layar yang digunakan. Saat ini yang terpopuler adalah layar normal dengan xhdpi. (yudha yudhanto, 2017).

2.3.2 Fitur-Fitur Android

Menurut (wadi, 2018), Android memiliki fitur-fitur yang menarik sehingga memikat hati para pengembang, selain fitur-fitur, sistem operasi ini juga memiliki kelebihan yang menjadi daya tarik. Berikut adalah kelebihan-kelebihan android.

1. Kode sumber terbuka (*open source*).
2. Siapapun bisa mengembangkan aplikasi nya.
3. Memiliki komunitas pengembang yang besar dan terus berkembang.
4. Memiliki pangsa pasar yang terus berkembang.
5. Memiliki resiko sukses lebih tinggi.
6. Memiliki lingkungan pengembangan (*development environment*) yang luas.
7. Kaya akan fitur-fitur.

Perangkat android ini mempunyai fitur perangkat keras di dalamnya di antaranya adalah :

1. *Touchscreen*, perangkat android memiliki fitur layar sentuh yang memberikan pengguna untuk berinteraksi dengan aplikasi menggunakan jari. Pengguna juga dapat menggunakan *swipe*, *flip*, *drag*, dan *pinch* untuk zoom. Android juga mendukung *multitouch* yang berarti keseluruhan layar dapat di sentuh.
2. GPS, sistem operasi android mendukung GPS yang memungkinkan developer untuk mengakses lokasi pengguna.
3. *Accelerometer*, android mendukung *accelerometer*, yaitu perangkat yang di gunakan untuk mengukur percepatan. *Accelerometer* dapat memberitahukan apabila suatu perangkat android bergerak, terguncang atau berbalik arah posisinya.
4. *SD Card*, android memiliki fitur yang kemungkinan pengguna (menyimpan atau membuka) file pada SD Card.
5. Google APis, sistem operasi android memungkinkan dengan tidak terbatas membuat panggilan telepon, mengorganisasikan kontak atau menginstal aplikasi.

Sebagai sistem operasi yang sangat populer Android mampu bersaing dengan sistem operasi Apple 4GS dan memiliki fitur fitur yang sangat luar biasa. Sebagaimana akan di tampilkan pada tabel dibawah ini.

Tabel 2.1 Fitur-Fitur Android

(wadi, 2018)

NO	FITUR	DESKRIPSI
1	<i>User Ineterface</i> (UI) yang menarik	Sistem operasi android memiliki UI yang menarik , interaktif dan intuitif bagi penggunanya.
2	Koneksifitas (<i>connectivity</i>)	Sistem Operasi Android mendukung koneksifitas seperti GSM/EDGE, IDEN,CDMA, EV DO, UMTS, Bluetooth, WIFI, LTE, NFC dan WiMAX.
3	Storage	Mendukung SQLate sebagai database relasional.
4	Media Support	Mendukung perangkat perangkat media seperti suara, citra , video .
5	Messagane	SSM dan MMS
6	Web Browser	Memiliki web Browser Chorme V 8 yang berbasis javascript dan mendukung HTML 5 dan CSS3
7	Multi touch	Memiliki fitur bawaan multi thouch
8	Multi Thasking	Fitur yang mengizinkan pengguna menjalankan beberapa aplikasi secara bersamaan.
9	Consutime Widget	Memiliki widget widget yang bisa di ubah
10	Mutli Language	Memiliki fitur yang mendukung beberapa bahasa dunia

11	Google Cloud Messaging (GCM)	GCM merupakan layanan yang mengizinkan pengembang untuk mengirimkan pesan /data ke pengguna perangkat android
12	Wifi	Mendukung fitur wifi yang terdapat pada perangkat mobile
13	Android Beam	Merupakan fitur berbasis teknologi NFC yang digunakan untuk berbagi secara instan.

2.3.3 Komponen Aplikasi Android

Aplikasi android ditulis dengan bahasa pemrograman Java. Semua file kode *intermediate* dan asset di satukan dalam suatu paket berupa file berekstensi. APK, sebuah file yang dapat didistribusi. Tiap file.APK, adalah sebuah aplikasi tunggal. Komponen aplikasi android terdiri dari beberapa jenis, antara lain: (Pertiwi, 2016).

1. Activity

Activity adalah istilah yang digunakan dalam pemrograman android untuk mengacu pada suatu interaksi dengan pengguna melalui antarmuka grafis (*graphical user-interface*, GUI). Sebagai satuan interaksi, *Activity* adalah tampilan yang anda lihat di layar seperti *windows* atau kotak dialog pada pemrograman aplikasi desktop. Tiap aplikasi dapat terdiri dari nol atau lebih *Activity*. Selain sebagai satuan interaksi dengan pengguna, *Activity* juga satuan eksekusi utama yang digunakan untuk memperbarui tampilan *user inter-face* (UI- Thread).

2. Intent

Intent adalah istilah yang digunakan dalam pemrograman android untuk mengacu pada mekanisme berbagi pesan pemberitahuan atau bertukar data *Activity* atau untuk menjalankan aplikasi lain.

3. *Service*

Service adalah komponen aplikasi yang berjalan di atas layar tanpa *user interface* untuk menyediakan layanan tertentu seperti mengecek RSS feed secara kontinu atau memainkan music. *Media player* adalah sebuah contoh aplikasi yang menggunakan *Service*.

4. *Content Provider*

Content Provider membuat suatu aplikasi dapat berbagi sejumlah data tertentu kepada aplikasi lain. Jika membutuhkan data nama-nama kontak, aplikasi anda tinggal meminta data tersebut.

5. *Broadcast Receiver*

Broadcast Receiver adalah komponen yang memantau, menerima, dan bereaksi terhadap pesan yang di sebar, baik oleh sistem maupun aplikasi lain. Misalnya, ketika baterai lemah, android akan mengirim pesan “baterai lemah” kepada semua *broadcast receiver* yang ingin di beritahu pesan ini. Untuk menggunakan *broadcast receiver*, pada dasarnya, anda hanya perlu membuat turunan tipe *broadcast receiver*, melengkapi metode *onReceive()*, dan mendaftarkannya di *androidManifest.xml* atau dengan metode *Context.registerReceiver()*. Instance *broadcast Receiver* hanya valid selama pemanggilan metode *onReceive()* sehingga *onReceive()* di panggil adalah mewakilkannya ke komponen lain.

2.4 Sejarah Java

Java di ciptakan oleh suatu tim yang di pimpin oleh Patrick Naughton dan James Gosling dalam suatu proyek dari Sun Microsystem yang memiliki kode Green dengan tujuan untuk menghasilkan bahasa computer sederhana yang dapat dijalankan di peralatan sederhana dengan tidak terikat pada arsitektur tertentu.

Mula mula James Gosling menyebut bahasa pemrograman yang di hasilkan nya dengan OAK, tetapi karena OAK sendiri merupakan nama dari bahasa pemrograman komputer yang sudah ada maka kemudian Sun mengubahnya menjadi Java.

Akhirnya setelah melalui beberapa transformasi dan proses. Sun akhirnya meluncurkan *browser* dari Java yang di sebut Hot Java yang mampu menjalankan applet. Setelah itu teknologi Java di adopsi oleh Netscape yang memungkinkan program java di jalankan di *browser* Netscape sejak Januari 1996 yang kemudian di ikuti oleh internet explorer. Karena keunikan dan kelebihanannya, teknologi java mulai menarik banyak vendor terkemuka seperti IBM, Symantec, Inprise, dll.

Akhirnya Sun meliris versi awal java secara resmi pada awal 1996 yang kemudian terus berkembang hingga muncul JDK 1.1 kemudian JDK 1.2 yang mengandung banyak peningkatan dan perbaikan sehingga mulai versi ini java di sebut Java 2. (Rickyanto, 2003).

2.4.1 Java

Java adalah perangkat lunak produksi Sun Microsystem Inc., yang merupakan perangkat lunak pemrograman untuk beberapa tujuan (*multi purpose*), *multiplatform* (dapat berjalan di beberapa sistem operasi), mudah di pelajari, dan *powerful*. Aplikasi aplikasi yang dapat di buat dengan Java, meliputi pemrograman web (*web programming*), pemrograman desktop (*desktop programming*), pemrograman handphone/ mobile (*mobile programming*) .saat ini java di akuisisi oleh Oracle Inc, yang membuat java semakin berkembang pesat dan populer (Supardi, 2014).

Perangkat lunak Java, sintakmnya (tulisanannya) mirip dengan C, karena bahasa java di buat mem akai bahasa pemrograman C. Pertama liris Java di sebut JDK (*Java Development Kit*), sehingga JDK versi 2 atau di kenal dengan **Java 2**, yang dibagi menjadi 3 edisi, yaitu **J2SE** (*Java Standard Edition*), **J2EE** (*Java 2 Enterprise Edition*), dan **J2ME** (*java 2 Micro Edition*).

1. J2SE merupakan edisi atau teknologi untuk pemrograman desktop atau aplikasi layar (*console*). J2SE merupakan perangkat lunak dasar yang harus di install sebelum anda memakai J2EE dan J2ME

2. J2EE merupakan edisi atau teknologi untuk pemrograman enterprise, seperti pemrograman database, JSP, Bean, dan lain lain.
3. J2ME merupakan edisi atau teknologi untuk pemrograman mobile/hanphone dan peralatan kecil (*small device*).

Setelah menginstal Java, di dalam sistem komputer anda terdapat **JVM** (*Java Virtual Machine*) . di dalam JVM tersebut terdapat **JRE** (*java Runtime Environment*). Program yang di ketik dengan bahasa java memiliki eksistensi **.java**, akan menghasilkan file class jika anda kompilasi. File kelas(.class) dapat di jalankan dengan memanfaatkan JRE.

2.5 Kotlin

Kotlin adalah bahasa pemrograman berbasis Java Virtual Machine (JVM) yang dikembangkan oleh JetBrains. Kotlin merupakan bahasa pemrograman yang pragmatis untuk android yang mengkombinasikan object oriented(OO) dan pemrograman fungsional. Kotlin juga bahasa pemrograman yang interoperabilitas yang membuat bahasa ini dapat digabungkan dalam satu project dengan bahasa pemrograman Java. Bahasa pemrograman ini juga dapat digunakan untuk pengembangan aplikasi berbasis desktop, web dan bahkan untuk backend. Beberapa keuntungan yang mungkin akan didapatkan jika pengembangan aplikasi beralih menggunakan Kotlin untuk mengembangkan aplikasi diatas platform JVM adalah sebagai berikut: (Sibarani, Munawar, & Wisnuadhi, 2018).

1. Dapat mengatasi NullPointerException yang umumnya terdapat pada Java
2. Penulisan kode lebih ringkas dan mudah dibaca dibandingkan kode yang ditulis dengan menggunakan bahasa Java
3. Mudah dipelajari
4. Dukungan IDE untuk mempermudah dalam pemrograman

2.6 Dart

Dart adalah bahasa berorientasi objek (*Object Oriented*) dengan sintaksis (*Syntax*) *C-Style* yang dapat di ubah secara opsional menjadi JavaScript. Mendukung berbagai macam alat bantu pemrograman seperti antar muka (*Interface*), *class*, *collection*, *generics*, dan opsional *typing*. Dart dapat digunakan untuk membuat aplikasi Web Android iOS dan juga menjalankan Server. Dengan menggunakan Dart dapat membuat UI (*User Interface*) yang indah dan berkualitas pada setiap *device* dengan menggunakan : (Syaputra dan Ganda, 2019) bab

1. Bahasa yang mengoptimalkan client

Dart pertama kali dioptimalkan untuk web apps dan berevolusi untuk membantu pengembangan *Mobile App*. Dart juga dapat kita gunakan untuk menjalankan Command Line juga Server-Side.

2. Kaya akan Framework

3. Tools yang Fleksibel dan menyenangkan

2.6.1 Tipe Data Pada Dart

Salah satu karakteristik paling mendasar dari bahasa pemrograman dart adalah himpunan tipe data yang dimiliki oleh bahasa pemrograman. Ini adalah jenis nilai yang dapat direpresentasikan dan dimanipulasi dalam bahasa pemrograman.

Dart memiliki tipe-tipe data berikut:

1. Numbers
2. String
3. Booleans
4. Lists
5. Maps

2.7 Java SE Development Kit (JDK)

Program android umumnya dibuat menggunakan bahasa Java. Oleh karena itu, anda akan membutuhkan *Java Development Kit* (JDK) untuk mengembangkan aplikasi android. JDK adalah piranti lunak yang anda perlukan untuk mengembangkan aplikasi java yang terdiri dari beberapa *utility* seperti compiler, interpreter, *Java Virtual Machine*, dan lain-lain. (P.Juhara, 2016).

2.8 Android SDK Platform

Menurut Rickyanto (2003), SDK adalah kependekan dari *Standart Development Kit* yang merupakan bekal utama bagi developer untuk membuat program dan menjalankan program java.

Android SDK di rancang untuk memudahkan anda berkerja dengan berbagai versi android dalam satu lingkungan pengembangan yang sama. Karena kebutuhan android berbeda tergantung pengembangan aplikasi dan untuk meminimalkan ukuran dan waktu *download*, bagian android SDK platform dan piranti yang terkait platform tersebut sengaja di pisah dari android SDK.

SDK platform berisi versi android yang menjadi target pengembangan aplikasi anda. Contoh, jika ingin membuat aplikasi dengan target Android 1.6 (*Donut*), anda harus *men-download* SDK platform API versi 4. Untuk menginstal android SDK platform, dari Eclipse IDE, pilih *Help-AVD SDK Manager*. Jika terhubung ke internet, *AVD SDK Manager* *men-download* daftar SDK platform yang tersedia. (Satyaputra dan Aritonang, 2014).

2.9 Android Development Tools (ADT)

Plugin *Android Development Tools* (ADT) untuk Eclipse adalah sebuah pengembangan lebih lanjut dari eclipse yang dapat di gunakan untuk mendukung pembuatan dan proses debugging dari aplikasi android yang sedang dibuat.

Dengan ADT, anda akan dapat melakukan hal-hal berikut ini dengan menggunakan eclipse. (Wahana Komputer, 2013).

1. Membuat proyek android baru.
2. Mengakses peralatan android emulator beserta tipe nya.
3. kompilasi dan debug aplikasi android.
4. Ekspor aplikasi android menjadi file paket APK.
5. Membuat sertifikasi digital untuk code-signing atas aplikasi APK yang telah anda buat.

2.10 Android Virtual Device (AVD)

AVD atau emulator adalah aplikasi komputer yang meniru cara kerja perangkat android sesungguhnya. Hampir sebagian besar perangkat android dapat anda temukan pada emulator, kecuali beberapa fitur seperti GPS, kamera atau sensor.

Emulator memungkinkan anda menguji dan melakukan debugging aplikasi dengan mudah dan cepat tanpa harus membawa piranti keras android. Karena berupa piranti maya, emulator dapat melakukan emulasi beragam perangkat keras dan versi android (Pertiwi, 2016).

2.11 Android Studio

Android studio adalah IDE (*Integrated Development Environment*) resmi untuk pengembangan aplikasi Android dan bersifat open source atau gratis. Peluncuran Android Studio ini diumumkan oleh Google pada 16 Mei 2013 pada *event* Google I/O Conference untuk tahun 2013. Sejak saat itu, Android Studio menggantikan Eclipse sebagai IDE resmi untuk mengembangkan aplikasi Android. Android studio sendiri dikembangkan berdasarkan IntelliJ IDEA yang mirip dengan Eclipse disertai dengan ADT plugin (Android Development Tools).

Fitur-fitur android studio : (Juansyah, A., 2015).

1. Proyek berbasis pada Gradle Build.
2. Refactory dan pembenahan bug yang cepat.

3. Tools baru yang bernama “Lint” dikalim dapat memonitor kecepatan, kegunaan, serta kompetibelitas aplikasi dengan cepat.
4. Mendukung Proguard And App-signing untuk keamanan.
5. Memiliki GUI aplikasi android lebih mudah.
6. Didukung oleh Google Cloud Platfrom untuk setiap aplikasi yang dikembangkan.

2.12 UML (*Unified Modeling Language*)

Menurut Windu dan Grace, (2013), “UML (*Unified Modeling Language*)” adalah bahasa spesifikasi standar yang di gunakan untuk mendokumentasikan, menspesifikan danmembangun perangkat lunak. UML merupakan Metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. (Suendri, 2018).

UML (*Unified Modeling Language*) adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualilasi, menspesifikasikan, membangun dan pendokumentasian dari sebuah sistem pengembangan *software* berbasis OO (*Object-Oriented*). UML juga memberikan standar penulisan sebuah sistem *blue print*, yang meliputi konsep bisnis proses , penulisan kelas kelas dalam bahasa pemrograman yang spesifik, skema *database*, dan komponen komponen yang di perlukan dalam sistem *software*.

2.12.1 Bangunan Dasar UML

Menurut Siti Fatimah,(2015) untuk memahami UML, kita membutuhkan model konseptual dari “bahasa”. Menggunakan 3 bangunan dasar untuk mendeskrikan sistem / perangkat lunak yang di kembalikan, yaitu:

1. Benda (*things*)
2. Relasi (*Relationship*)

3. Diagrams

Setiap bangunan dasar dapat di terapkan sepanjang tahap pengembangan sistem. Ketiga jenis bangunan dapat digunakan secara saling melengkapi satu sama lain (contoh lebih mendalam akan dijelaskan pada bagian bagian selanjutnya).

Secara umum UML di terapkan dalam pengembangan sistem atau perangkat lunak berorientasi objek sebab metodologi ini umumnya memiliki keunggulan sebagai menggambarkan class tertentu (satu *class* dapat memiliki lebih dari satu *statechart* diagram). Diagram UML (*Unified Modeling Language*) (Suendri,2018). antara lain sebagai berikut:

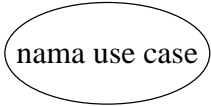
1. Use Case Diagram



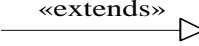
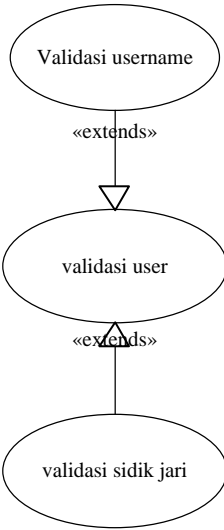
Use Case diagram di gunakan untuk menggambarkan sistem dari sudut pandang pengguna *User*. Sehingga pembuatan *use case* diagram lebih di titik beratkan pada fungsionalitas yang ada pada sistem, bukan berdasarkan alur atau urutan kejadian. Sebuah *use case* diagram mempresentasikan sebuah interaksi anta aktor dengan sistem.


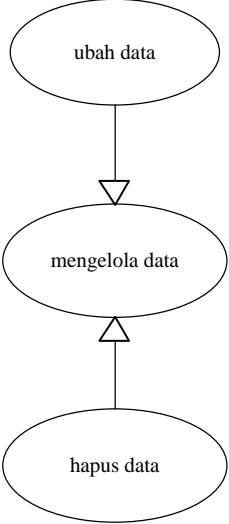
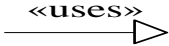
Menurut (Rosa A.S, 2016), Berikut adalah simbol-simbol yang ada pada diagram *use case*;

Tabel 2.2 Simbol Use Case Diagram

(Rosa A.S, 2016)

Simbol	Nama	Deskripsi
	Use case	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal di awal frase nama <i>use case</i>
Aktor / actor	Aktor / actor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem yang akan dibuat itu sendiri, jadi walaupun

 nama aktor		simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor
	Asosiasi/ association	Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor
	Ekstensi/ extend	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan, misal</p>  <pre> graph TD A([Validasi username]) -- «extends» --> B([validasi user]) C([validasi sidik jari]) -- «extends» --> B </pre> <p>Arah panah mengarah pada <i>use case</i> yang ditambahkan; biasanya <i>use case</i> yang menjadi extend-nya</p>



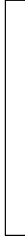
		merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya
	Generalisasi / generalization	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya, misalnya</p>  <p>Arah panah mengarah pada <i>use case</i> yang menjadi generalisasinya (umum)</p>
	Menggunakan / include / uses	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini

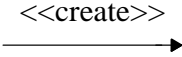
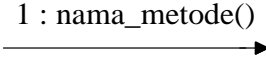
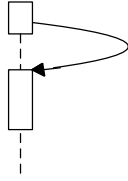
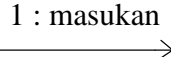
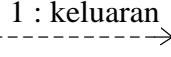
2. Sequence Diagram

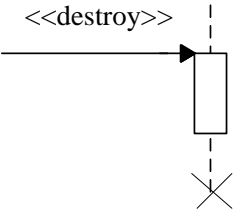
Menggambarakan interaksi antara sejumlah objek dalam urutan waktu. Kegunaannya untuk menunjukkan rangkain pesan yang di kirim antara objek juga interaksi antar objek yang terjadi pada titik tertentu dalam eksekusi sistem.

Tabel 2.3 Simbol Sequence Diagram

(Rosa A.S, 2016)

Simbol	Nama	Deksripsi
 nama aktor <div style="border: 1px solid black; padding: 2px; width: fit-content; margin-top: 10px;"> <u>nama_aktor</u> </div>	Aktor Atau Tanpa waktu aktif	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu orang; biasanya dinyatakan menggunakan kata benda di awal frase nama actor
	Garis hidup / lifeline	Menyatakan kehidupan suatu objek
<div style="border: 1px solid black; padding: 2px; width: fit-content;"> <u>nama objek : nama kelas</u> </div>	Objek	Menyatakan objek yang berinteraksi pesan
	Waktu aktif	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya, misalnya</p> <pre> sequenceDiagram participant A A->>A: 1: login() activate A A->>A: 2:cekStatusLogin() A->>A: 3:open() deactivate A </pre> <p>Maka cekStatusLogin() dan open() dilakukan didalam metode login ()</p>

		Aktor tidak memiliki waktu aktif
Pesan tipe create 		Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
Pesan tipe call 	Pesan tipe call	Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri,  z Arah panah mengarah pada objek yang memiliki operasi / metode, karena ini memanggil operasi / metode maka operasi / metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi
	Pesan tipe send	Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim
	Pesan tipe return	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian




	Pesan tipe destroy	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy
---	--------------------	---



3. Activity Diagram

Menggambarkan rangkaian aliran dari aktivitas, di gunakan untuk mendeskripsikan aktifitas yang di bentuk dalam suatu operasi sehingga dapat juga di gunakan untuk aktifitas lainnya. Diagram ini sangat mirip dengan flowchart karean memodelkan *workflow* dari suatu aktifitas ke aktifitas yang lainnya, atau dari aktifitas ke status. Pembuatan *activity diagram* pada awla pemodelan prosese dapat membantu memahami keseluruhan proses. *Activity diagram* juga di gunakan untuk menggambarkan interaksi antara bebeapa *use case*.

Tabel 2.4 Simbol Activity Diagram

(Rosa A.S, 2016)

Simbol	Nama	Deskripsi
	Stastus awal	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah awal
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
	Percabangan/ decision	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu

	Penggabungan / join	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
	Status akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
<div data-bbox="322 860 821 1173"> <p>nama swimlane</p> </div> <p>Atau</p> <div data-bbox="322 1252 807 1624"> <p>nama swimlane</p> </div>	Swimlane	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

2.12.2 Tujuan *Unified Modelling Language* (UML)

Tujuan UML diantaranya adalah :

1. Memberikan model yang siap pakai, bahasa pemodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.

2. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa. ‘
3. Menyatukan praktek-praktek terbaik yang terdapat dalam pemodelan. (Zufria, 2013).

2.12.3 Tool-Tool yang mendukung *Unified Modelling Language* (UML)

Saat ini, banyak sekali tool pendesainan yang mendukung UML, baik itu *tool* komersial maupun *opensource* (Zufria, 2016). Beberapa di antaranya adalah:

1. *Rational Rose*
2. *Together*
3. *Object*
4. *Jvision*
5. *Objecteering*
6. *magicDraw*
7. *Visual Object Modeller*
8. *Visual Paradigm for UML*

2.13 HIPO

HIPO merupakan alat dokumentasi program yang berdasarkan fungsinya untuk meningkatkan efisiensi usaha perawatan program. Dokumen ini dilaksanakan dengan mempercepat lokasi dalam kode pada fungsi program yang akan dimodifikasi. Atau dapat dikatakan bahwa HIPO dikembangkan agar tersedia suatu teknik untuk mendokumentasikan fungsi program. Pembentukan HIPO ini dilakukan pada tahap pengembangan sistem informasi. (Rian, 2014).

Adapun tujuan HIPO adalah sebagai berikut :

1. Untuk memberikan struktur yang memungkinkan fungsi suatu sistem dapat dimengerti.

2. Untuk menguraikan fungsi-fungsi yang akan dikerjakan oleh suatu program, bukan untuk mengkhususkan pernyataan program program yang di pakai untuk melaksanakan fungsi-fungsi tersebut.
3. Untuk memberikan deskripsi visual dari input yang akan dipakai serta output yang akan dihasilkan oleh masing-masing fungsi pada tiap-tiap tingkat diagram.

Tujuan HIPO yang paling penting adalah untuk menghasilkan output yang benar dan dapat memenuhi kebutuhan user.

Jenis Diagram dalam Paket HIPO :

1. Daftar Isi Visual/ Visual Tabel of Contents (VTOC)

Terdiri dari satu diagram hirarki atau lebih. Visual tabel of contents menggambarkan seluruh program HIPO baik rinci maupun ringkasan yang terstruktur. Pada diagram ini nama dan nomor dari program HIPO diidentifikasi. Struktur paket diagram dan hubungan fungsi juga diidentifikasi dalam bentuk hirarki. Keterangan masing- masing fungsi diberikan pada bagian penjelasan yang diikutsertakan dalam diagram ini.

2. Diagram Ringkasan/ Overview Diagram

Suatu seri diagram fungsional. Masing-masing diagram di hubungkan dengan salah satu fungsi sistem. Diagram ringkasan menggambarkan fungsi dan referensi utama dari sistem. Fungsi dan referensi ini diperlukan program untuk memperluas fungsi sampai uraian yang terkecil. Diagram ini berisi input, proses dan output dari fungsi khusus. Input pada diagram ini berisis item-item data yang dipakai oleh proses, sedangkan proses merupakan urutan langkah-langkah yang menelaskan fungsi yang sedang dijalankan untuk menghasilkan suatu output. Output berisikan item-item data yang dihasilkan dan diubah oleh proses. Anak panah pada diagram ringkasan dari input ke proses menunjukkan hubungan antara item data di input dan langkah-langkah proses, sedangkan anak panah

dari proses ke output menunjukkan hubungan tahap-tahap proses dan item data output. Penjelasan tentang tahap-tahap proses, item data input dan output diberikan pada bagian extended description.

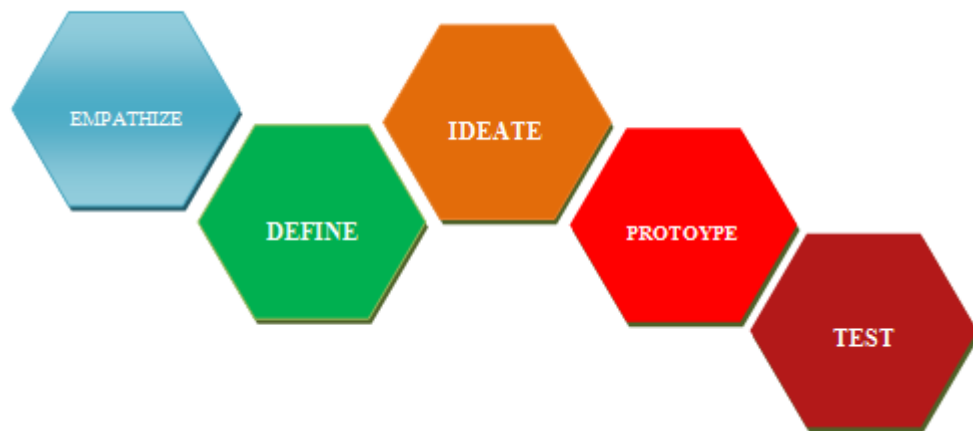
3. Diagram Rinci/ Detail Diagram

Suatu seri diagram fungsional dan masing-masing diagram dihubungkan dengan sebuah sub-fungsi sistem diagram rinci merupakan diagram yang paling rendah dalam diagram yang terdapat dalam paket HIPO. Diagram rinci berisi unsur-unsur paket dasar. Fungsi dari diagram ini adalah menjelaskan fungsi-fungsi khusus, menunjukkan item-item output dan input yang khusus dan menunjukkan diagram rinci lainnya. (Rian H., 2014)

2.14 *Design Thinking*

Dalam melakukan penelitian ini, metode yang di gunakan dalam melakukan perancangan adalah metode “*Design Thinking*” metode ini di kenal sebagai suatu proses berpikir komprehensif yang berkonsentrasi untuk menciptakan solusi. Menurut (hussein, 2018). *Design Thinking* merupakan alat yang di gunakan dalam *problem-Solving*, *problem design* , hingga *problem –forming*. Tidak hanya untuk m enyelesaikan suatu permasalahan , namun juga untuk membentuk dan merancang suatu permasalahan. Solusi yang di ciptakan dari metode “*Design Thinking*” terdiri dari 5 tahapan antara lain : (Brown & Kazt, 2009).

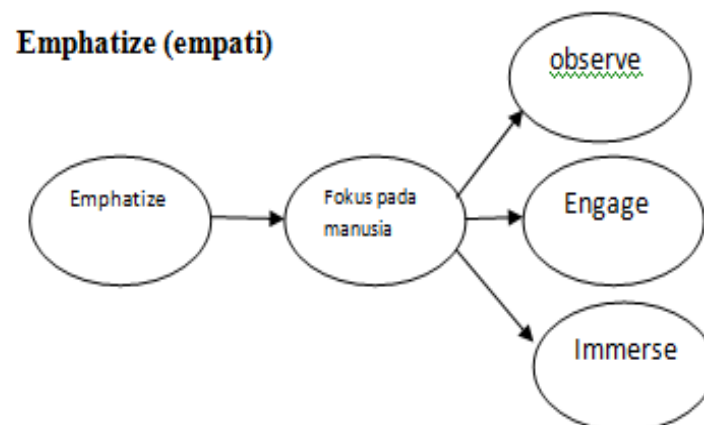
Design Thinking : A 5 Stage Process



Gambar .2.10 Diagram tahapan dalam Motode Design Thinking

2.14.1 *Emphatize*

Tahapan yang pertama ada *Emphatize* (empeti) yang di anggap sebagai inti dari proses perancangan yang berpusat pada manusia , metode ini berupaya untuk memahami pengguna dalam konteks produk yang akan di rancang, dengan melakukan observasi, wawancara, dan menggabungkan observasi dan wawancara dengan terlebih dahulu di berikan suatu skenario



Gambar 2.11 Tahapan *Emphatize*

1. *Observe* (amati)

Mengamati apa yang di lakukan *user* dan cara mereka berinteraksi dengan lingkungannya membantu kalian untuk memahami kebutuhan mereka.

2. *Engage* (terlibat)

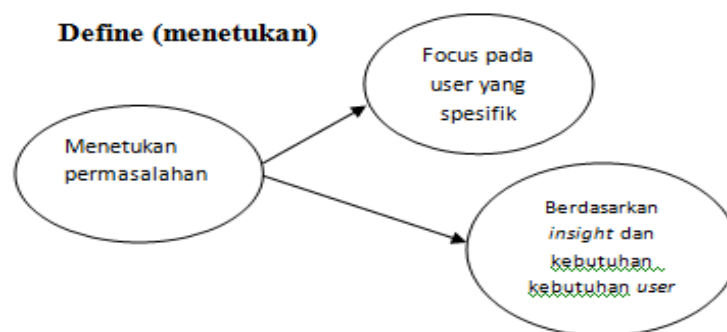
Terlibat dengan orang secara langsung dapat membantu dalam mengungkapkan cara berpikir dan nilai nilai yang mereka pagang. Kita dapat megerti pandangan *user* melalui cerita dan hal hal yang mereka lakukan.

3. *Immerse* (rasakan langsung)

Setelah mengamati dan melibatkan diri dengan *user*, penting untuk merasakan langsung pengalaman *user*. Temukan (atau jika perlu ciptakan pengalaman tersebut agar kalian bisa merasakan situasi yang dialami oleh *user*).

2.14.2 *Define*

Tahapan yang kedua *Define* (menentukan) yang merupakan proses menganalisis dan memahami berbagai wawasan yang telah di peroleh dalam empati, dengan tujuan sebagai untuk menentukan pertnyataan masalah sebagai *point of view* atau perhatian utama pada penelitian.



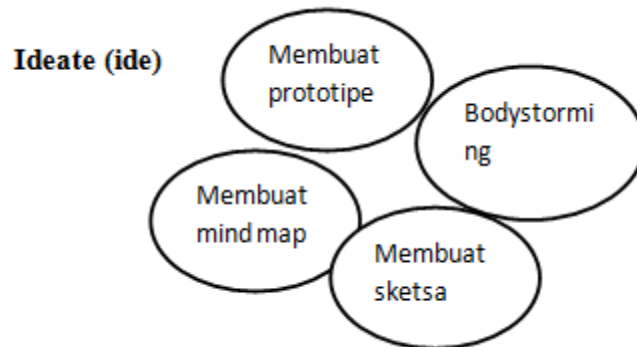
Gambar 2.12 Tahapan *Define*

2.14.3 *Ideate*

Tahapan yang ketiga *ideate* (ide) yang merupakan proses transisi dari rumusan masalah ke penyelesain masalah, adapun dalam proses *ideate*

ini akan berkonsentrasi untuk menghasilkan gagasan atau ide sebagai landasan dalam membuat prototype rancangan yang akan di buat.

- Mengeluarkan ide ide yang dapat menjadi solusi terhadap permasalahan.
- Cara menghasilkan ide



Gambar 2.13 Tahapan *Ideate*

2.14.4 *Prototype*

Tahapan yang ke empat *Prototype* di kenal sebagai rancangan awal suatu produk yang akan di buat ,untuk mendeteksi kesalahan sejak dini dan memperoleh berbagai kemungkinan baru. Dalam penerapannya , rancangan awal yang di buat akan di uji coba kepada pengguna untuk memperoleh respond dan *feedback* yang sesuai untuk menyempurnakan rancangan .

2.14.5 *Test*

Tahapan yang kelima adalah *test* (Uji coba) atau pengujina yang di lakukan untuk mengumpulkan *feedback* pengguna dari berbagai rancangan akhir yang telah di rumuskan dalam protitipe sebelumnya. Proses ini merupakan tahap akhir namun bersifat *life cycle* sehingga memungkinkan perulangan dan kembalai pada tahap peranangan apabila terdapat kesalahan (Razi, Mutiaz, & Setiawan, 2018)..

Alasan melakukan test :

- a. Memperbaiki prototype.

- b. Mempelajari *user* lebih dalam.
- c. Mengetes dan mempernaiki pernyataan masalah.

2.15 Pengertian Zakat dan Dalil Hukum

Zakat menurut bahasa artinya tumbuh, bersih, atau menambah kebaikan, sebagaimana firman Allah SWT dalam surah *at-taubah* (9):103 :

خُذْ مِنْ أَمْوَالِهِمْ صَدَقَةً تُطَهِّرُهُمْ وَتُزَكِّيهِمْ بِهَا وَصَلِّ عَلَيْهِمْ إِنَّ صَلَاتَكَ سَكَنٌ لَهُمْ وَاللَّهُ سَمِيعٌ عَلِيمٌ

Ambilah zakat dari sebagian harta mereka, dengan zakat itu kamu membersihkan dan menyucikan mereka, dan mendoalah untuk mereka. Dan Allah Maha Mendengar lagi Maha Mengetahui .

Zakat secara istilah ialah mengeluarkan sebagian dari harta benda atas perintah Allah sebagai sedekah wajib kepada mereka yang telah ditetapkan menurut syariat – syariat yang telah ditentukan oleh hukum Islam sebagaimana firman Allah SWT dalam surah *at-taubah* (9) :60 . (Nurhayati dan Imran A.S, 2017).

Sesungguhnya zakat zakat itu , hanyalah untuk orang-orang fakir, orang-orang miskin, pengurus-pengurus zakat, para mukalaf yang dibujuk hatinya, untuk mereka yang sedang dalam perjalanan, sebagai suatu ketetapan yang di jadikan Allah, dan Allah Maha Mengetahui lagi Maha Bijaksana.

Zakat adalah ibadah *maliyah ijtima'iyah* (ibadah yang berkaitan dengan ekonomi keuangan dan kemasyarakatan dan merupakan salah satu dari rukun Islam yang mempunyai status dan fungsi yang penting dalam syariat Islam, sehingga Alquran menegaskan kewajiban zakat bersama dengan kewajiban sholat di 82 tempat.. (Zuhdi M, 1993).

Adapun menurut Sabiq (2006), zakat merupakan nama dari sesuatu hak Allah yang dikeluarkan kepada fakir miskin. Dinamakan zakat karena mengandung harapan untuk mendapat berkah, membersihkan, dan memupuk jiwa dengan kebaikan.

Tujuan utama di wajibkan zakat atas umat islam itu adalah untuk memecahkan problem kemiskinan, pemeratakan pendapatan, dan meningkatkan kesejahteraan umat dan negara. Dan tujuan ini tidak akan tercapai, apabila pelaksana zakat di serahkan sepenuhnya kepada kemauan para wajib zakat. Demikian pula kalau zakat di kelola oleh badan badan amil zakat nonpemerintah yang jumlahnya tidak terbatas jumlahnya tanpa pengawasan, pengendalian dan pembinaan pemerintah.

2.15.1 Harta Yang Wajib Dizakati

Menurut Nurhayati dan Imran (2017) Mahmud Syaltut mengemukakan, Al Quran dan hadis masing masing memberikan informasi tentang harta yang wajib di zakati. Kedua nash yang berlainan dan bersamaan itu harus di pandang sebagai informasi yang *integrative* dan *kondusif* satu dengan yang lainnya. Pernyataan Syaltut itu di setuju pula oleh Yusuf Qardawi dengan mengemukakan beberapa jenis kekayaan yang di sebut Al-quran, yaitu “

1. Emas dan perak dalam firman allah SWT, yang artinya : “*Orang-orang yang menimbun emas dan perak dan tidak membelanjakannya pada jalan allah sampaikanlah kepada mereka berita gembira tentang azab yang sangat pedih*” (QS.at-taubah (19):34).
2. Tanaman dan buah buahan yang dinyatakan allah ,” *makanlah sebagian buahnya bila berbuah dan bayarlah hak tananman itu ketika hak memanennya*” (QS.al-An’aam(16):141).
3. Usaha misalnya usaha dagang dan lain lain firman allah ,yang artinya “*Hai orang orang yang beriman, keluarkanlah sebagian yang baik dari hasil usahamu*”. (QS. Al-Baqarah (2):267).
4. Barang-barang tambang yang dikeluarkan dari hasil perut bumi. Allah Berfirman.”*Dan berikanlah sebagian dari apa yang kami keluarkan dari bumi untuk kamu*” (QS. al-Baqarah (2);267).

Sementara itu, rasullah SAW menginformasikan harta yang wajib dizakati yang terbatas pada logam mulia (emas dan perak), hewan ternak

(unta, lembu, kambing), tanaman (biji gandum), dan buah buahan (kurma dan anggur). Persoalan ini terus bergulir hingga menjadi kesepakatan ulama dan adapula yang berijtihad bahwa selain itu dapat di-*qiyas*-kan pada harta benda lain. Demikian menurut penuturan dari Syaltut. (Nurhayati dan Imran, 2017).

2.15.2 Jenis Harta Yang Tidak Dizakati

Islam memerintahkan kepada para pemeluknya agar bekerja keras mencari rejeki yang halal guna mencukupi kebutuhan hidup dirinya dan keluarganya, baik kebutuhan jasmani maupun kebutuhan rohaninya (perhatikan Al-Qur'an Surat Al-Mulk ayat 15 dan surat Al-Jum'ah ayat 10). Islam memberikan kebebasan kepada setiap individu Muslim memilih jenis usaha, pekerja atau profesi yang sesuai dengan bakat, keterampilan, kemampuan, atau keahliannya masing masing, baik yang berat dan kasar yang memberikan penghasilan kecil (*blue color*) seperti tukang becak, maupun yang ringan dan halus yang mendatangkan penghasilan besar (*white color*) seperti notaris. (Zuhdi M, 1993).

Tidak semua harta benda atau kekayaan yang dimiliki oleh seseorang terkena zakat. Ada beberapa hal yang perlu diperhatikan antara lain:

1. Bebas zakat, seperti rumah tempat tinggal beserta isinya, mobil pribadi, dan peralatan kerja;
2. Wajib dizakati seperti harta benda nya saja, seperti emas dan perak , apabila telah mencapai nisab dan haul nya;
3. Wajib dizakati penghasilan dari harta benda dan penghasilan yang timbul daripadanya, seperti hasil dari tanah pertanian/perkebunan, dan sewa gedung;
4. Wajib zakat dizakati harta benda dan penghasilan yang timbul daripadanya, seperti hasil dari peternakan sapi dan perdagangan.

Adapun jenis penghasilan yang terkena zakat berdasarkan ketentuan Al-Qur'an dan hadis Nabi adalah sebagai berikut :

Tabel 2.5 Pembagian zakat
(Zuhdi M, 1993)

No	Jenis Usaha	Nisab	Haul/waktu	Persentase	Dasar Hukum
1	Perdagangan	93,6 gr	Setahun	2,5%	Hadis nabi
2	Pertanian/perkebunan (beras, jagung, gandum, kurma, dan angur)	750 kg	Waktu panen	5% dengan irigasi 10% tadah hujan	Al-Baqarah ayat 267 dan hadis nabi
3	Perternakan a. Unta b. Sapi/kerbau c. Kambing	5 ekor 30 ekor 40 ekor	Setahun Setahun Setahun	Seekor kambing biasa 2 tahun lebih. Seekor anak sapi/kerbau umur 2 tahun lebih Seekor kambing betina biasa umur 2 tahun lebih, atau seekor kambing domba betina umur 1 tahun lebih	Hadis Nabi

2.15.3 Menghitung Zakat Binatang Ternak

Jumhur ulama sependapat bahwa binatang ternak, yaitu unta sapi, dan kambing wajib di zakati, sedangkan syarat wajib zakat binatang ternak adalah : (Nurhayati dan Imran A, S;, 2017).

1. Sampai senisab.
2. Berlangsung setahun.
3. Binatang ternak tersebut di gembalakan rumput untuknya.
4. Tidak di pekerjakan untuk kepentingan pemiliknya, seperti di pekerjakan untuk menggarap tanah pertanian, di jadikan alat untuk mengambil air untuk menyirami tanaman, di gunakan untuk alat angkut barang barang dan sebagiannya.

Sesuai dengan *ijma'* ulama dan hadis-hadis yang bersumber dari Rasulullah SAW dan sahabat, maka nisab dan kadar zakat unta, sapi, dan kambing dapat di lihat sebagai berikut: (Nurhayati dan Imran A.S, 2017).

1. Unta nisabnya dari 5-120 ekor

- a. 5-9 ekor zakatnya 1 ekor kambing umur 2 tahun.
- b. 10-14 ekor zakatnya 2 ekor kambing umur 2 tahun.
- c. 15-19 ekor zakatnya 3 ekor kambing 3 tahun.
- d. 20-24 ekor zakatnya 4 ekor kambing umur 2 tahun.
- e. 25-35 ekor zakatnya 1 ekor unta umur 1 tahun.
- f. 36-45 ekor zakatnya 1 ekor unta umur 2 tahun.
- g. 46-60 ekor zakatnya 1 ekor unta umur 3 tahun.
- h. 61-75 ekor zakatnya 1 ekor unta umur 4 tahun.
- i. 76-90 ekor zakatnya 2 ekor unta umur 2 tahun.
- j. 91- 120 ekor zakatnya 3 ekor unta umur 3 tahun.
- k. 121-.... Ekor zakatnya 3 ekor unta umur 2 tahun.

Mulai dari 121 ekor ini di hitung tiap tiap 40 ekor untuk zakatnya 1 ekor unta umur 2 tahun dan tiap tiap 50 ekor unta zakatnya 1 ekor unta umur 3 tahun.

2. Sapi atau lembu nisabnya 40-500 ekor

- a. 30 -39 ekor zakatnya sapi jantan umur 1 tahun yang memasuki tahun kedua (*tabi'*).
- b. 40-59 ekor zakatnya sapi betina umur 2 tahun memasuki tahun ketiga (*musinnah*).
- c. 60-69 ekor 2 *tabi'*.
- d. 70-79 ekor zakatnya 1 *tabi'* dan 1 *musinnah*.
- e. 80-89 ekor zakatnya 2 *musinnah*.
- f. 90-99 ekor zakatnya 3 *tabi'*.
- g. 100-... ekor zakatnya 2 *tabi'* dan 1 *musinnah*.

3. Kambing nisabnya 40-500 ekor

- a. 40-120 ekor zakatnya 1 ekor kambing.
- b. 121-200 ekor zakatnya 2 ekor kambing.
- c. 201-399 ekor zakatnya 3 ekor kambing.
- d. 400-499 ekor zakatnya 4 ekor kambing.
- e. 500-599 ekor zakatnya 5 ekor kambing.

Seterusnya, setiap 100 ekor kambing di tambah zakatnya 1 ekor kambing

2.15.4 Menghitung Zakat Profesi atau Penghasilan

Pada zaman sekarang banyak sekali pekerjaan dan profesi yang mendatangkan jumlah uang yang cukup tinggi di bandingkan dengan hasil panen seorang petani yang yang wajib zakat. demi keadilan yang di junjung tinggi, maka status sosial yang tinggi di masyarakat muslimin karena jumlah uang yang banyak diwajibkan mengeluarkan zakat dari hasil pendapatannya. Pekerjaan yang menghasilkan uang banyak banyak yang di maksud biasanya ada dua jenis. (Rochim, 1434 H), Yaitu:

1. Berkat kecekatan dan otak (professional) seperti penghasilan dokter spesialis, insinyur, advokat, seniman, penjahit, kontraktor(pengembangan), dan lain lain.

2. Pekerjaan yang di kerjakan seseorang buat pihak lain baik pemerintahan ataupun pengusaha dengan memperoleh honorarium dan gaji.

Cara perhitungan zakat profesi

1. Model memperoleh harta penghasilan (profesi) mirip uang, dengan panen (hasil pertanian), sehingga harta ini dapat di kiaskan pada zakat pertanian berdasarkan nisabnya senilai 520 kg beras (653 kg padi/gabah = lima ausaq) dengan waktu pengeluaran zakatnya (setiap kali panen)/ setiap kali mendapatkan gaji atau penghasilan.
2. Model harta yang di terima sebagai penghasilan berupa uang, sehingga jenis harta ini dapat di kiaskan pada zakat harta (simpanan atau kekayaan) berdasarkan kadar zakat yang harus di bayarkan (2,5%). Dengan demikian, apabila hasil profesi seseorang telah memnuhi ketentuan wajin zakat, ia bekewajibab menunaikan zakatnya.

Rumus :

$A = \text{Harga beras} \times \text{Nisab (520 kg beras)}..$

$B = (\text{Penghasilan perbulan} + \text{Pendapatan,bonus dll perbulan} - \text{Pengeluaran kebutuhan perbulan}) \times 2.5 \%$

Jika penghasilan A lebih besar dari B maka wajib membayar zakat..

Contoh menghitung zakat profesi:

Pak abdul baqi adalah seseorang kariawaan swasta yang berdomisil di medan, ia mempunyai istri dan dua orang anak yang masih kecil. Penghasilan perbulannya adalah Rp10.000.000

Pendapatan gaji perbulan Rp 10.000.000

Bonus perbulan 1.000.000

Pengeluaran perbulan 5.000.000

Nisab 520 kg beras @Rp 10.000 (relative) Rp. 5.200.000,-

$(10.000.000 + 1.000.000 - 5.000.000) \times 2.5 \%$

Zakat yang harus di tunaikanRp 150.000,-

2.15.5 Zakat Fitrah

Zakat fitrah diwajibkan atas setiap orang muslim yang merdeka yang mempunyai kelebihan makanan pada dirinya dan bagi orang-orang yang nafkahnya menjadi tanggungannya, yang banyaknya cukup untuk menunaikan zakat fitrah tersebut. Apabila jumlahnya hanya sebagian dari seluruh zakat fitrah yang wajib dikeluarkan, maka ada pendapat yang mengatakan, ia tetap harus mengeluarkannya, dan ada pula pendapat yang mengatakan tidak.

Barang siapa wajib untuk mengeluarkan zakat fitrah untuk dirinya, maka wajib pula atasnya mengeluarkan zakat fitrah orang-orang yang menjadi tanggungannya. Kalau mereka semua muslimin, dan ia cukup untuk mengeluarkan zakat fitrah dari semuanya. Jika ia hanya bisa mengeluarkan zakat fitrah hanya untuk sebagian saja, maka hendaklah ia memulainya dengan orang yang wajib diberinya nafkah. Ada pendapat lain bahwa ia harus mendahulukan fitrah istrinya dari pada dirinya sendiri. Dan ada pendapat lain mengatakan bahwa ia boleh memilih untuk dirinya dan untuk yang lainnya.

Zakat fitrah mulai diwajibkan di akhir bulan Ramadhan, dan matahari sudah terbenam. Demikian menurut qoul yang paling shahi, sedangkan menurut qoul lain, wajib setelah terbit matahari pada hari berikutnya. Yang paling utama adalah mengeluarkannya sebelum sholat "ied". Dan boleh dikeluarkan pada hari-hari sepanjang bulan Ramadhan. Tetapi tidak boleh ditunda sampai habis hari raya 'Idul Fitri, barang siapa mengakhirinya maka berdosa lah ia, dan ia harus mengqadhanya.

Banyaknya zakat fitrah yang wajib dikeluarkan tiap orang adalah satu sha', sesuai dengan sha' Rasulullah shallallahu alaihi wa sallam yaitu 3.1 liter. Yang wajib dikeluarkan itu adalah makanan yang wajib dizakati, seperti : kurma, anggur kering, gandum, beras, dan yang serupa. Dan wajib dikeluarkan zakat fitrah dari makanan pokok yang berasal dari suatu negeri tersebut. (Abdullah H, 1992)

Zakat fitrah pelaksanaan pembagiannya adalah di prioritaskan untuk fakir miskin karena maksud utamanya adalah untuk membantu fakir miskin pada hari lebaran, zakat fitrah di keluarkan untuk setiap orang sebanyak 2.5 kg beras. Hal ini juga dapat di keluarkan dalam bentuk uang. (Daradjat Z, 1994). Akan tetapi dalam mazhab imam syafi'I bahwa mengeluarkan kadar zakat fitrah lebih dari 2,5 kg itu lebih baik, maka rata-rata masyarakat Indonesia yang mengikuti mazhab imam syafi'I mengeluarkan kadar zakatnya adalah 2.7 kg beras.

1. Hukumnya

Zakat fitrah itu subah *wajibah* bagi setiap pribadi kaum muslimin. Ini sesuai dengan riwayat Abdullah bin Umar as , bahwa rasullah SAW mewajibkan zakat fitrah di bulan ramadhan berupa satu sha' kurma atau satu sha' gandum terhadap budak dan orang yang merdeka, laki laki dan perempuan , kecil dan dewasa dari kalangan kaum muslimin . (HR.an-Nas'i)

2. Kadaranya, waktu mengeluarkanya, dan orang yang berhak menerimanya

Kadar zakat fitrah itu adalah satu sha', yaitu empat mud yang di keluarkan dari makanan pokok penduduk negeri .ia bisa anda keluarkan sebelum menunaikan sholat idul fitri. Jika anda mengeluarkannya setelahnya, maka hukumnya sama dengan sedekah biasa saja.

Sedangkan mengenai orang yang berhak menerimanya, maka hal itu sama dengan yang lainnya, yaitu delapan golongan yang di sebutkan allah SWT dalam Al-quran Al-karim, hanya para fakir dan miskin lebih di utamakan ; sesuai dengan sabda Rasulullah SAW yang artinya:

“cukupkanlah mereka dari meminta minta pada hari ini” (HR.al-Baihaqi).

2.15.6 Zakat fitrah uang atau makanan

Menurut Para ulama sepakat bahwa zakat fitrah boleh di keluarkan dalam bentuk makanan pokok. Ada tiga pendapat ulama menanggapi pembagian zakat fitrah antara lain. (Abdullah H, 1992).

1. Di bolehkan membayar zakat fitrah dan yang lainnya menggunakan qimah (mata uang). Ini merupakan pendapat ulama hanafiyyah.
2. Tidak di bolehkan membayar zakat dengan qimah (mata uang). Pendapat ini di nyatakan oleh ulama Syafi'iyyah dan Hanabilah.
3. Di bolehkan membayar zakat dengan qimah (mata uang) bila terdapat di dalamnya maslahat.berdasarkan buku "*buku panduan zakat*".

Kemaslahatan membayar zakat dalam bentuk uang pada saat ini merupakan sesuatu yang tidak bisa di pungkiri. Kebutuhan mustahik sangat beragam, tidak hanya bahan makanan pokok .kadang kala memeberikannya dengan bahan makanan pokok justru merugikan penerima zakat. Sebab, untuk memenuhi kebutuhan yang ia harus menjual lagi harta zakat yang ia terima dengan harga di bawah standar.

Adapun tatacara perhitungan dalam aplikasi yang akan dirancang adalah :

$(\text{Harga beras saat ini} \times 2.7 \text{ kg Beras}) \times \text{Jumlah orang yang membayar zakat} .$

Contoh :

Pak ahmad memiliki istri dan 2 orang anak dan harga beras yang dikonsumsi oleh keluarga pak ahmad ialah 12.000/kg.

Maka :

$$(12.000 \times 2,7) \times 4 = 129.600$$

Jadi

Pak ahmad harus membayar zakat senilai Rp 129.600.

2.15.7 Zakat Emas

Istilah zakat mal identik dekat harta kekayaan berupa tabungan uang, perdagangan, maupun emas dan perak. Nisab zakat emas , perak uang dan perdagangan sesuai dengan 85 gr emas. Emas yang menjadi standar adalah emas murni .nilai zakat yang dikeluarkan sebesar 2,5 %. Seperti contoh ataupun ilustrasi di bawah ini: (Rochim, 1436 H).

Bapak ahmad memiliki emas sebanyak 100 gr .nilai itu sesungguhnya telah mencapai nisab emas, setelah satu tahun. Cara menghitung zakatnya, yaitu:

Harga emas saat ini = Rp 500.000.

Nisab emas = $500.000 \times 85 = \text{Rp } 42.500.000$

Rumus

(Jumlah emas yang telah mencapai nisab x harga emas saat ini) x 2,5 % .

$(100 \text{ gr Emas} \times 500.000) \times 2.5 \% = \text{Rp } 1.250.000.$

Jadi zakat emas yang wajib dikeluarkan bapak ahmad adalah 1.250.000.

2.15.8 Golongan Orang Yang Menerima Zakat

Ulama sepakat ada delapan golongan orang yang menerima zakat berdasarkan firman allah SWT ,”*sesungguhnya zakat itu hanyalah untuk orang orang fakir, orang orang miskin, orang pengurus-pengurus zakat, para mualaf yang di bujuk hatinya, untuk (memerdekakan) budak, orang orang yang berhutang, untuk di jalan allah dan orang orang yang sedang dalam perjalanan sebagai sesuatu ketetapan yang di wajibkan allah. Dan allah maha mengetahui dan maha bijaksana* “(QS.9:60). Dan penjelasan lebih jauh tentang delapan golongan tersebut adalah sebagai berikut:

1. Orang orang fakir
2. Miskin

Para ulama berbeda pendapat tentang pengertian fakir miskin. Sebagian ulama berpendapat bahwa fakir adalah orang yang tidak memiliki pekerjaan dan tidak memiliki harta yang mencukupi kebutuhan hidupnya. Sedangkan miskin adalah orang yang memiliki harta dan penghasilan tetapi belum mencukupi pokok hidupnya .menurut Hanafiyyah, orang fakir miskin adalah orang yang tidak memiliki harta yang mencapai nisab.

3. Pengurus zakat atau amil. Pada zaman sekarang Rasulullah SAW, beliau mengutus sebagian sahabat untuk menarik zakat lalu beliau memberikan upah kepada mereka sebagai ganti waktu dan tenaga yang telah mereka korbakan.

4. Mualaf ialah orang yang baru masuk islam dan kondisi nya lemah.
5. Budak, yang memerdekakan dirinya.
6. Orang yang berutang. Yang di maksud dengan orang yang berutang disini adalah orang yang tidak memiliki harta sama sekali untuk membayar hutang yang jatuh tempo. Jika ia memiliki harta , harta itu hanya mampu untuk mencukupi kebutuhanpokoknya saja.
7. Fi sabilillah. Sebagian besar ulama tafsir berpendapat bahwa kata “fi sabilillah” berarti berperang di jalan allah. Dalam berbagai forum zakat, ulama kontenforer menguatkan pendapat yang menunjukkan fi sabilillah adalah jihad. Dalam hal ini pengertian jihat tidak sebatas pada peperangan, tetapi dapat berupa tenaga, fisik, tulisan, dan lisan .
8. Ibnu sabil. Ibnu sabil adalah seseorang yang kehabisan perbekalan saat dalam perjalanan. Sebagina besar ulama mensyaratkan perjalan tersebut bukanperjalan untuk kemaksiatan. (Rochim, 1436 H).

2.16 Fidyah Puasa

Puasa dalam al-Quran dan Hadis di sebut dengan kata *ash-shaum*, dan secara harifah berarti (menahan diri dari sesuatu). Menurut istilah agama islam, puasa adalah “menahan diri dari lapar, minum, hubungan seksual, dan segala yang membatalkan, mulai dari terbit fajar terbenam matahari, dengan niat karena allah SWT.

Manusia dalam banyak hal perlu memiliki kemampuan menahan diri .orang yang tidak mempunyai kemampuan menahan diri adalah orang yang negative, dan puasa berguna untuk menumbuhkan kemampuan menahan diri sehingga dapat memperkuat kepribadian seseorang. Dengan demikian, ketika seseorang melaksanakan puasa (tidak melakukan perbuatan), sesungguhnya di dalamnya terdapat makna yang positif.

Arti fidyah menurut *lughah* (bahasa) ialah tebusan atau penebusan sedangkan menurut istilah artinya penebusan yang wajib di lakukan seseorang karena sesuatu hal ia tidak dapat melakukan kewajibannya seperti Puasa Ramadhan dan dalam pelaksanaan ibadah haji. (Daradjat Z, 1994)

Firman Allah dalam Al-Quran, QS Al-Baqarah, 2: 183-184.

Artinya

“hai orang-orang yang beriman. Diwajibkan atas kamu berpuasa sebagaimana diwajibkan atas orang-orang sebelum kamu agar kamu bertaqwa. (yaitu) dalam beberapa hari tertentu. Maka jika di antara kamu ada yang sakit atau dalam perjalanan (lalu ia berbuka) wajiblah baginya berpuasa sebanyak hari yang ditinggalkan itu pada hari-hari yang lain. Dan wajib bagi orang-orang yang berat menjalankan (jika mereka tidak berpuasa) membayar fidyah yaitu (member makan orang miskin). Maka barang siapa yang dengan kerelaan hati mengerjakan kebajikan maka itulah yang lebih baik baginya. Dan berpuasa lebih baik bagi kamu jika kamu mengetahui.”

2.16.1 Hakikat puasa

Hakikat puasa adalah menahan diri dari segala yang di larang oleh agama terhadap seluruh anggota badan kita, baik lahir maupun batin, untuk itu, ada lima hal yang berkaitan dengan puasa, (Shalih S.I.,2011) yaitu:

1. Menjaga pandangan mata dari segala hal yang tercela;
2. Menjaga lisan agar tidak berkata dusta, tidak mengatakan perkataan kotor, tidak *ghibah* dan lain-lain.
3. Menjaga telinga agar seluruh suara atau ucapan yang di benci tidak di dengarkan.
4. Menjaga perut agar jangan ada makanan yang haram masuk ke dalamnya dan tidak berlebih-lebihan dalam berbuka.
5. Sehabis berbuka, melakukan intropeksi diri, memikirkan apakah ibadah puasanya pada hari itu di terima Allah atau tidak. Bahkan, intropeksi diri ini dapat di terapkan setiap selesai melakukan suatu ibadah, baik ibadah *mahdhah* maupun ibadah *ghairu mahdhah*.

2.16.2 Sebab mengganti puasa

Qadha puasa tidak harus di kerjakan secara langsung , segera sesudah ramadhan. Akan tetapi boleh di tunda , sebagaimana yang tuturkan oleh Aisyah alaihi salam, *“aku mempunyai tanggungan utang puasa Ramadhan , tetapi aku tidak mampu membayarnya kecuali pada bulan Sya’ban.”*

Jika engkau terbebani utang puasa ramadhan, lalu engkau menundanya hingga sampai bulan Ramadhan tahun berikutnya. Maka kamu tetap wajib membayar hutang puasa itu pada bulan berikutnya, yaitu mulai bulan Syawwal ,sebagaimana firman allah terhadapmu. Apabila engkau sudah berbuka pada hari pertama bulan Syawwal, maka usahakan membayar hutang hari hari puasa yang terbebani atas dirimu. Kamu tidak berkewajiban memberi makan orang miskin atau lainnya , karena tidak ada syariat yang mewajibkanmu member makan orang miskin pada keadaan ini, berlandaskan hadist marfu’ yang sampai kepada Nabi SAW.

Di bawah ini paparan mengenai wanita haid dan nifas dalam fiqih yang tidak wajib untuk berpuasa pada saat bulan Ramadhan, (Shalih S.I.,2011).

1. Wanita Haid dan Nifas Tidak wajib Puasa

Ulama sepakat bahwa wanita yang haid atau nifas tidak boleh berpuasa , harus berbuka di bulan ramadhan dan menggantinya di hari lain, dan jika keduanya tetap maka hal itu tidak sah, Aisyah berkata :” kami haid pada zaman rasullah SAW , lalu kami di perintahkan mengganti puasa dan tidak mengganti sholat.”(HR *Muttafaq’alaih*). Dan perintah ini datang dari rasullah SAW.

Wanita haid dan nifas sama hukumnya , karena darah nifas sama dengan darah haid, begitu juga hukumnya, kapan saja di temukan darah haid pada bagian dari siang hari maka puasa menjadi rusak pada hari itu, baik di awal hari atau sore hari . jika seorang wanita haid berniat puasa, padahal ia tahu tidak boleh mengerjakan puasa maka ia berdosa dan tidak sah.

2. Gugurnya Kewajiban Puasa bagi Wanita Hamil dan menyusui

Ulama sepakat bahwa wanita hamil dan menyusui mendapat rukhsah jika ia khawatir terhadap dirinya, dan wanita menyusui jika mengkhawatirkan anaknya. Hal tersebut dikarenakan syarat *qudrah* (mampu) tidak ada di dalam diri keduanya, dan merujuk pada keumuran firman Allah SWT:

Barang siapa dari kalian yang sakit atau dalam perjalanan maka hendaknya ia menggantinya di hari hari lain. (QS.AL-Baqarah (2):184

Al-Fakhr Ar-Razi menjelaskan maksud dari kata “sakit” di dalam ayat di atas, “sakit adalah tidak bekerjanya fungsi tubuh secara baik dalam melakukan pekerjaan secara normal atau yang sesuai.”

Mereka kemudian berpendapat mengenai sakit yang membolehkan tidak puasa ke dalam tiga arus pendapat yaitu sebagai berikut :

Pertama, orang sakit apa saja atau musafir dengan tujuan apa saja boleh mengambil rukhsah tidak berpuasa. Ini berpendapat Al-Hasan dan Ibnu Sirin,.

Kedua, rukhsah ini hanya untuk yang sakit yang seandainya ia berpuasa akan mengalami kesulitan dan keberatan.

Ketiga, dan ini merupakan mayoritas ulama , sakit yang membolehkan seseorang berbuka adalah sakit yang membahayakan diri dan memperparah penyakit , sebab tidak ada perbedaan dalam perbuatan antara yang dikhawatirkan dengan yang menyebabkan rasa takut.

2.16.3 Orang yang membayar fidyah (kaffarat)

1. Menurut para ulama, sanksi bagi mereka jika mereka menagguhkan qadha hingga tiba bulan Ramadhan berikutnya maka orang tersebut wajib membayar puasa sekaligus membayar kafarat. Ini pendapat Imam Maliki, Imam Asy-Syafi’I, dan Imam Ahmad.

2. Jika seseorang melakukan hubungan badan karena lupa ataupun sengaja kalau ia sedang berpuasa, maka kaffarat berlaku untuk nya dan qadha puasa. (Rusyd, 2010).
3. Jika seorang wanita hamil atau wanita yang sedang menyusui merasa takut kalau mereka puasa akan membahayakan diri mereka, maka merasa boleh tidak puasa , dan wajib atas mereka mengqadha. Dan jika mereka menguatirkan keadaan anak mereka, maka mereka boleh tidak puasa, dan wajib qadha. Dalam hal fidyah, ada tiga qoul:
 Pertama, wajib atas keduanya mengeluarkan fidyah tiap-tiap hari sebanyak satu mud makanan pokok.
 Kedua, mengeluarkan fidyah itu hanya sunnah.
 Ketiga, mengeluarkan fidyah itu hanya wajib atas wanita menyusui, tidak atas wanita hamil. (Abdullah, 1992).
4. Para ulama sepakat bahwa seorang nenek kakek yang udzur mereka berdua boleh berbuka. Mereka berdua harus membayar fidyah, besarnya fidyah yang harus di bayar adalah satu *mud* setiap harinya.

BAB III

METODE PENELITIAN

Bab ini menjelaskan metodologi penelitian yang digunakan untuk membangun aplikasi zakat dan fidyah. Adapun metodologi yang di gunakan dalam penelitian ini ialah metode *design thinking*.

3.1 Tempat dan Waktu Pelaksanaan

1. Tempat Penelitian

Tempat pelaksanaan penelitian ini adalah dilingkungan kampus Universitas Islam Negeri Sumatera Utara Medan di Jalan. IAIN kota Medan.

2. Waktu Penelitian

Waktu yang digunakan dalam penelitian ini, telah dimulai sejak bulan April untuk mempersiapkan data-data yang akan dikumpulkan, hingga sampai pemeliharaan sistem pada bulan Agustus 2019.

Tabel 3.1 Jadwal Penelitian

Jadwal Penelitian	April				Mei				Juni				Juli				Agustus			
	M I	M II	M III	M IV	M I	M II	M III	M IV	M I	M II	M III	M IV	M I	M II	M III	M IV	M I	M II	M III I	M IV
Persiapan																				
Proposal Skripsi																				
Penjadwalan																				
Seminar Proposal																				

Pengumpulan Data																			
Analisis Data																			
Studi Pustaka																			
Perancangan Sistem																			
Desain Interface																			
Pembuatan Coding																			
Testing																			
Pemeliharaan Sistem																			

Adapun tahap yang dilakukan dalam penelitian ini sebagai berikut :

1. Persiapan

Adapun tahap yang dilalui dalam penelitian ini adalah tahap persiapan, disini peneliti mencari masalah yang ada di tengah masyarakat serta solusi yang akan dijadikan objek penelitian.

2. Pengajuan proposal skripsi

Pengajuan proposal skripsi ini adalah tahap dimana seorang peneliti mengajukan judul skripsi kepada pihak prodi sebagaimana syarat-syarat yang telah ditentukan.

3. Penjadwalan

Penjadwalan ini dilakukan untuk mendapatkan sasaran yang tepat dalam penelitian sehingga memerlukan jadwal yang tersusun secara sistematis.

4. Seminar proposal

Setelah judul yang telah diajukan kepada pihak prodi, tahap ini adalah tahap yang menentukan apakah kasus yang ingin diangkat oleh seorang peneliti layak untuk dijadikan bahan penelitian atau tidak, dengan menyertakan seminar proposal.

5. Pengumpulan data

Setelah judul penelitian diterima, maka peneliti akan mengumpulkan banyak data serta sumber-sumber yang di perlukan secara study literatur, *observe* dan lain- lain.

6. Analisis data

Pada tahap ini peneliti, melakukan analisa terhadap semua data yang telah didapat, memilah-milih keakuratan serta keaslian data yang bagus untuk bahan skripsi.

7. Studi pustaka

Pada tahap ini peneliti melakukan studi pustaka yang di bimbing oleh dosen pembimbing 1 dan pembimbing 2 untuk menghindari kecurangan data dan plagiat selama penelitian berlangsung.

8. Perancangan sistem

Pada tahap ini peneliti membuat rancangan sistem dengan pemodelan uml untuk mendapatkan hasil yang sistematis mengenai alur sistem yang akan di rancang.

9. Desain *interface*

Setelah membuat rancangan sistem, peneliti mulai mendesain *interface*.

10. Pembuatan *coding*

Pada tahap ini peneliti, membuat kode pemrograman pada aplikasi perhitungan zakat dan fidyah puasa.

11. *Testing*

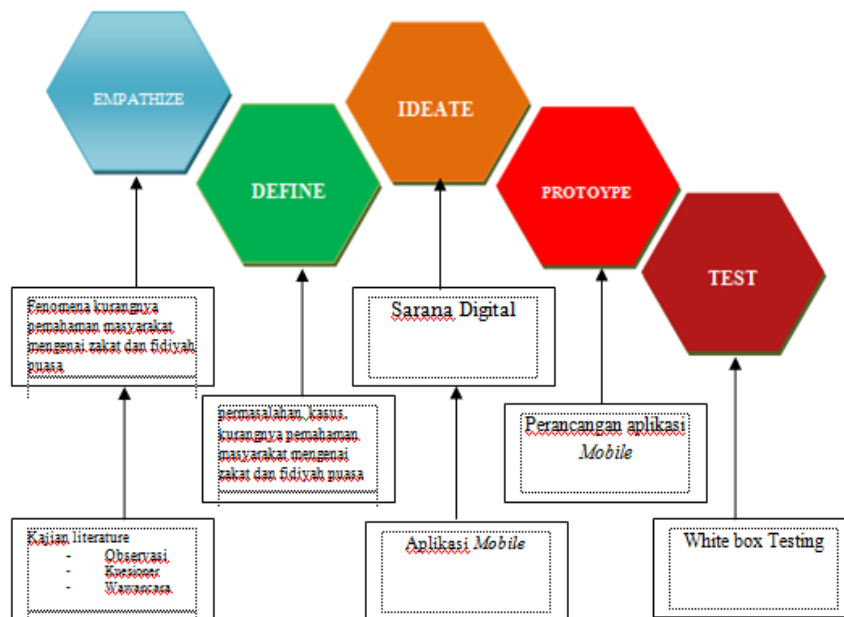
Setelah pembuatan kode pemrograman selesai, maka peneliti melakukan testing dengan menggunakan metode *white box*.

12. Pemeliharaan sistem

Pada tahap akhir ini peneliti, melakukan pemeliharaan sistem dimana, peneliti memeriksa kembali aplikasi yang telah jadi, sehingga peneliti mengetahui apakah aplikasi ini siap untuk digunakan masyarakat atau tidak.

3.2 Cara Kerja

Dalam melakukan penelitian ini, metode yang digunakan dalam melakukan perancangan adalah metode “*Design Thinking*” metode ini di kenal sebagai suatu proses berpikir komprehensif yang berkonsentrasi untuk menciptakan solusi.



Gambar 3.1 Diagram flow Metode Design Thinking.

3.2.1 Emphasize

Tahap ini peneliti melakukan pengamatan riset data mengenai jumlah, pembagian serta perhitungan zakat dan fidyah puasa yang sesuai dengan Undang-Undang syariat islam, serta mengetahui alur dari sistem yang berjalan. Guna membuat skenario yang efektif untuk mengembangkan sistem yang akan di rancang.

3.2.1.1 Observe (amati)

Terkait dengan besaran jumlah zakat yang dikeluarkan pada setiap masyarakat maka peneliti melakukan pengamatan pada masjid Al-Musannif dan lingkungan kampus UINSU untuk mengambil data yang akurat mengenai objek yang akan diteliti.

3.2.1.2 Wawancara

Pada tahap ini peneliti mewawancarai 2(dua) narasumber untuk mendapatkan data yang lebih akurat pada sumber yang terpercaya mengenai zakat dan fidyah antara lain sebagai berikut :

Wawancara ini dilakukan pada :

Hari : Kamis

Tanggal : 8 Agustus 2019, 5 Desember 2019

Narasumber : Bapak Ali Akbar Simbolon, M.Pd., Sugihartono

Jabatan : Dosen Universitas Islam Negeri Sumatera Utara, Nazir Mesjid Musannif.

Hasil : mengetahui tatacara perhitungan zakat dan fidyah puasa, membenarkan bahwa banyak masyarakat yang belum mengetahui aturan dalam zakat dan fidyah, mengetahui sistem yang sengan berjalan tatacata pembagian zakat secara manual.

3.2.1.3 Engage (terlibat)

Dengan adanya teknik *Engage*, peneliti terlibat secara langsung pada masyarakat hingga mengetahui permasalahan tentang objek yang diteliti. Adapun metode yang dilakukan untuk mengetahui permasalahan masyarakat mengenai objek tersebut, dengan melakukan penyebaran kuesioner dan Tanya jawab kepada masyarakat yang bersangkutan. Guna menghasilkan *Insight* pada tahap *Immerse* dalam metode *Design Thinking*.

Tabel 3.1 Data Masyarakat Yang Mengisi Kuesioner

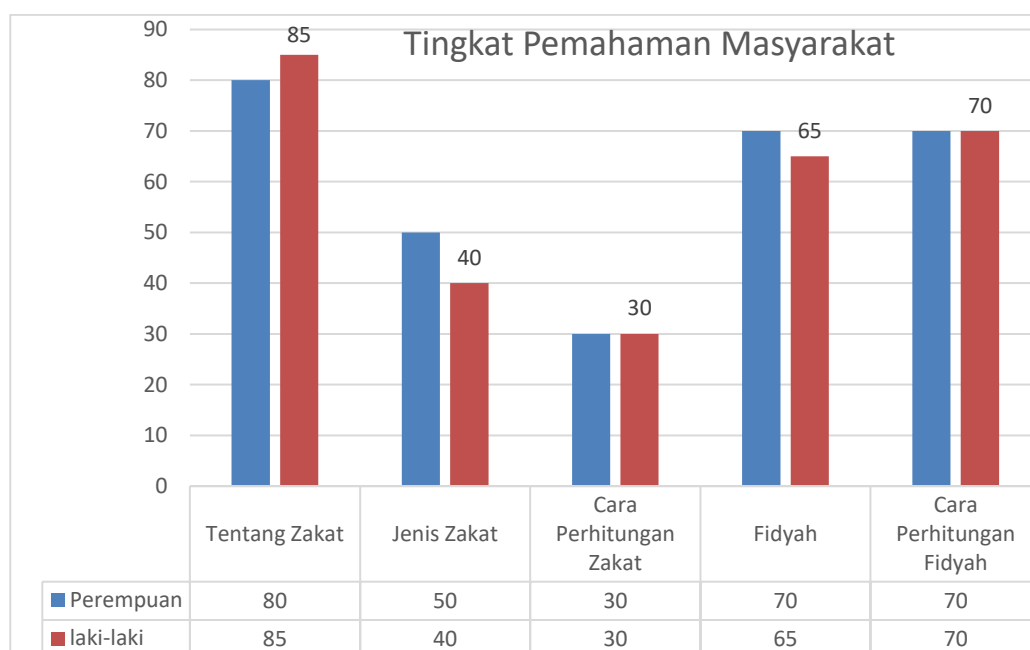
Nama	Pekerjaan	Universitas	Tempat Tinggal
Samauati Nur	Mahasiswa	UMA	Namorambe
Puspa Sari	Siswi	-	Jl.Psi No 1
Dekta	Shodow Teacher	-	Asrama UMA

Chika Anggraini	Ibu Rumah Tangga	-	Jl. Tuasan
Maulida Hayati	Ibu Rumah tangga	-	Jl. Tuasan
Dede Astika	Mahasiswa	UMA	Asrama Uma
Tri Suci	Mahasiswa	UMA	Jl. PBSI
Uni	Mahasiswa	UMA	Jl. PBSI
Guslila Sari	Mahasiswa	UINSU	Jln pancing
Herawati Hasibuan	Mahasiswa	UINSU	Lot dendang
Febry Dwi Syafitri	Mahasiswa	UINSU	Medan denai
Zahra Azura	Mahasiswa	UINSU	Bijai
Siti Afzalena	Mahasiswa	UINSU	Teladan
Rr Balqis Neureta	Mahasiswa	UINSU	Deli tua
Suci Umayrah	mahasiswa	UINSU	Ringroat
Dwiki Guntara	Mahasiswa	UINSU	Deli tua
Rio Fahrezi	Mahasiswa	UINSU	Bromo
Aulia Pratama	Mahasiswa	UINSU	Medan belawan
Habib Asy Muhyi	Mahasiswa	UINSU	Lot dendang
M. Isya Nasution	Mahasiswa	UINSU	Lot Dendang
M. Arif Suhada	Mahasiswa	UINSU	Berandan
Anindita Ayuningtias	Mahasiswa	UINSU	Marelan
Husna Hasibuan	Mahasiswa		Medan tembung

Irma Yunita	Mahasiswa		Lubuk pakam
-------------	-----------	--	----------------

3.2.1.4 *Immerse* (rasakan langsung)

Setelah melakukan tahap pengamatan, wawancara dan *engage* maka peneliti menghasilkan grafik terhadap permasalahan objek yang diteliti. Antara lain sebagai berikut:



Gambar 3.2 Grafik *Immerse*

3.2.1.5 Study Literatur

Pada tahap ini peneliti mengambil data atau sumber-sumber yang berhubungan dengan topik yang akan diteliti, dari berbagai sumber seperti buku, jurnal, dan situs situs internet. Seperti pembahasan mengenai android dari buku “*Mudah Membuat dan Berbisnis Aplikasi Android dengan Android Studio*” oleh (yudha yudhanto, 2017)Yudhanto & Wijayanto (2014. Pemodelan UML oleh Suendri (2018), Vol.03, Jurnal Ilmu Komputer dan Informatika. Penelitian terdahulu mengenai aplikasi zakat oleh Rifa’atunnisa, dkk (2014), Vol. 11, jurnal Algoritma Sekolah Tinggi

Teknologi Garut. Metode *Design Thinking* oleh Aria Ar razi, dkk (2018), Vol. 03, Jurnal Desain Komunikasi Visual, Manajemen Desain dan Periklanan.

3.2.2 Define (menentukan)

Hasil yang telah diperoleh dalam *emphatize*, Sebagai *point of view* atau perhatian utama pada penelitian tersebut. Peneliti menemukan beberapa permasalahan diantaranya ialah :

1. Rendahnya pemahaman masyarakat mengenai zakat dan fidyah puasa.
2. Masyarakat tidak mengetahui cara pembagian atau takaran zakat yang akan di keluarkan.
3. Masyarakat masih menghitung zakat dan fidyah secara manual.
4. Memanfaatkan sarana aplikasi *mobile* untuk menghitung jumlah zakat dan fidyah secara akurat.

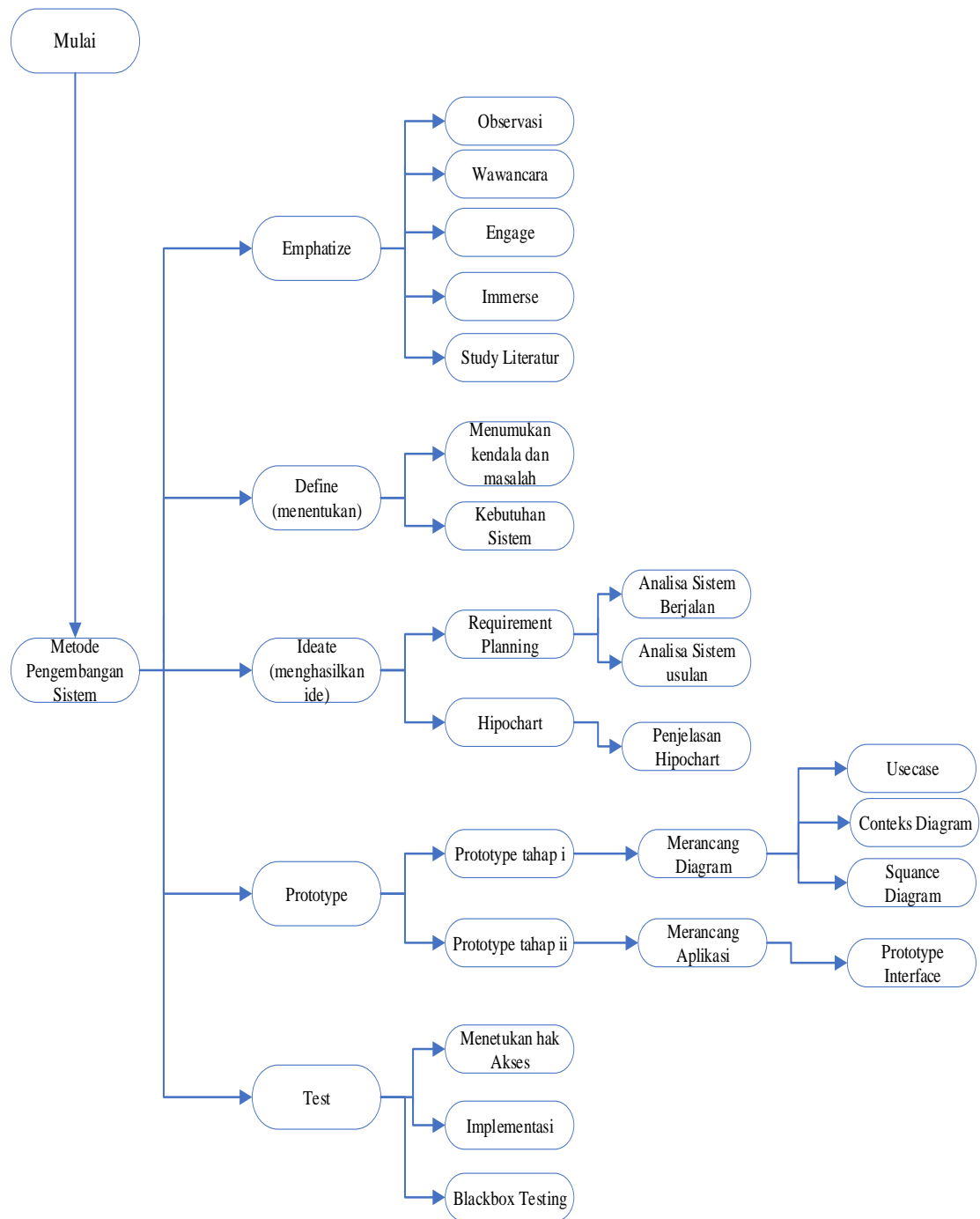
3.2.2.1 Kebutuhan Sistem

Dalam pengkodean aplikasi, untuk membangun sistem ini dilakukan dengan menggunakan :

1. Bahasa pemrograman Java.
2. Perangkat Keras:
 - a. AMD A9-9425 RADEON R5, 5 COMPUTE CORES 2C+3G (2 CPUs), ~3,1GHz.
 - b. Hardisk 1 Tb,
 - c. Memory 4GB,
 - d. Perangkat keras lainnya (*keyboard, mouse*, dan lain-lain).
3. Perangkat Lunak :
 - a. Windows 10 Pro 64-bit(10.0,Build 18362).
 - b. Microsoft Visio sebagai alat bantu untuk mendesain sistem berjalan dan sistem usulan.
 - c. Draw.io sebagai alat bantu untuk mendesain diagram dan rancangan antarmuka.

- d. Android Studio versi 3.5 untuk membangun aplikasi.
 - e. Browser internet Mozilla Firefox sebagai alat bantu untuk mencari berbagai referensi dan sumber.
4. Spesifikasi *smartphone*
- a. Android Versi 7.1.1 media untuk menjalankan aplikasi
 - b. Processor Eight core
 - c. Ram 3,00GB.
 - d. Memory Perangkat 32,0 GB.

3.2.3 Kerangka Berpikir



Gambar 3.3 Kerangka berpikir

BAB IV

HASIL DAN PEMBAHASAN

4.1 *Ideate* (ide)

Dari hasil *define* dan *emphitize* dalam *design thinking*, tahap yang ketiga adalah *Ideate*, dimana peneliti menghasilkan (ide) cara yang efektif serta efisien untuk meminimalisir permasalahan yang dihadapi di tengah-tengah masyarakat. Dengan memanfaatkan teknologi *mobile*, peneliti merancang sebuah aplikasi *mobile* untuk menghitung zakat dan fidyah, melalui aplikasi ini *user* hanya perlu menginput nilai atau jumlah yang akan di hitung pada aplikasi tersebut. Serta mengurangi ketidakpastian sasaran, jumlah zakat yang akan diberi (dikeluarkan). Aplikasi yang akan di rancang juga didukung dengan fitur informasi sehingga memudahkan dan menambah wawasan pengguna.

4.1.1 *Requirement Planning*

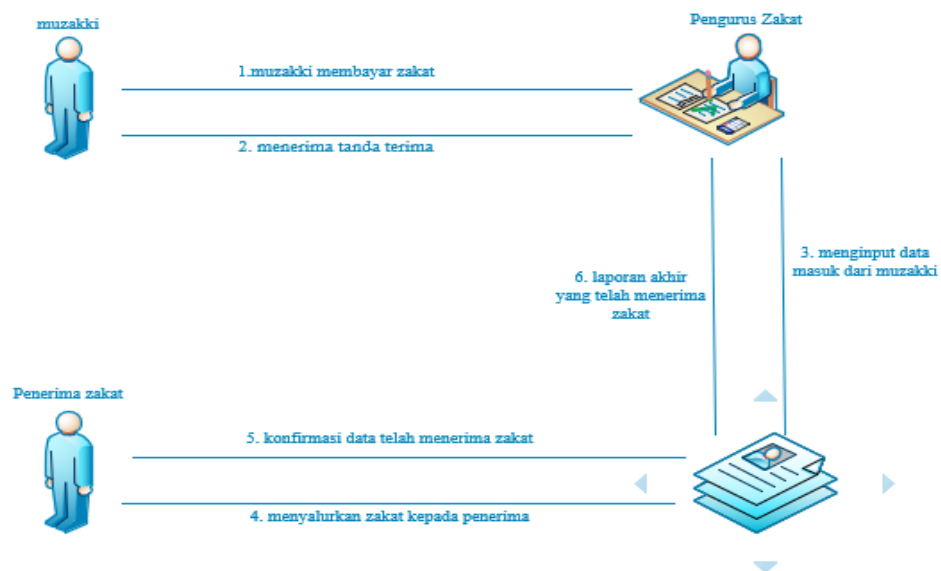
Requirement Planning atau fase perencanaan memiliki beberapa tahapan yang akan dibahas dan dijelaskan pada penelitian ini. Menganalisa sistem yang sedang berjalan, menganalisa sistem usulan dan membuat perancangan dari sistem yang telah diusulkan.

4.1.1.1 Analisa Sistem Berjalan

Berdasarkan pengamatan dan observasi yang telah di lakukan oleh peneliti berikut ini adalah alur dari pada sistem yang berjalan pada masyarakat sekitar.

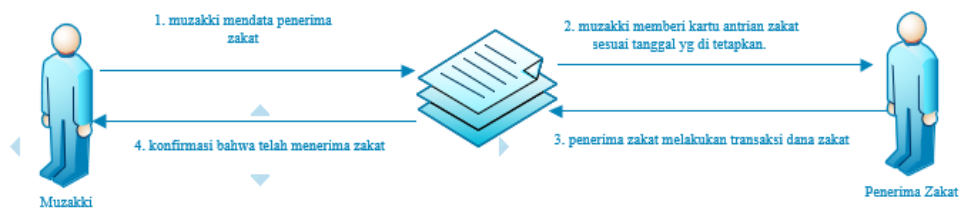
4.1.1.2 Gambaran Analisis Sistem Berjalan

Membayar zakat dengan menggunakan badan Amil Zakat (Melalui Lembaga Mesjid)



Gambar 4.1 Sistem Berjalan Pada Mesjid

a. Menyalurkan zakat secara langsung kepada masyarakat sekitar.



Gambar 4.2 Sistem Berjalan Pada Masyarakat

4.1.1.3 Narasi Sistem Berjalan

Pada sistem yang sedang berjalan ada dua cara seorang muzakki dalam membayar masjid sebagai berikut :

1. Sistem berjalan membayar zakat dari lembaga masjid :

Pada sistem ini seorang muzakki membayar zakat kepada lembaga masjid atau lembaga yang menyediakan tempat pembayaran zakat, di sini seorang

muzakki di haruskan mengisi beberapa data seperti identitas dan lain lain sebagainya. Kemudian muzakki menerima bukti transaksi bawa ia telah membayar zakat, lalu para pengurus zakat mendata (input dan update data) setiap muzakki yang telah membayar zakat serta memberikan laporan kepada pimpinan agar zakat dapat di salurkan kepada masyarakat (fakir miskin serta orang yang membutuhkan).

2. Sistem pembayaran zakat secara langsung

Pada sistem ini muzakki mendata setiap masyarakat di sekitar untuk menyalurkan dana zakat, dalam pendataan tersebut muzakki mengkalkulasi harta dan pendapatan, pengeluaran biaya pokok, hutang piutang dan lain-lain, kemudian setelah hasil akhir di peroleh muzakki memberikan kartu antrian zakat kepada penerima. Kartu antrian ini berfungsi untuk mengambil dana zakat sesuai hari yang telah di tentukan.

4.1.1.4 Identifikasi Masalah Sistem Berjalan

Berdasarkan sistem berjalan tersebut, peneliti mengidentifitaksi beberapa masalah diantaranya adalah :

1. Melalui lembaga masjid
 - a. Masyarakat tidak mendapatkan informasi tentang zakat.
 - b. Kurangnya wawasan tambahan mengenai zakat dan cara perhitungannya.
 - c. Perhitungan yang masih manual
2. Membayar zakat secara langsung
 - a. Muzakki melakukan perhitungan secara manual.
 - b. Tidak menambah wawasan mengenai zakat, selain jenis zakat yang di berikan

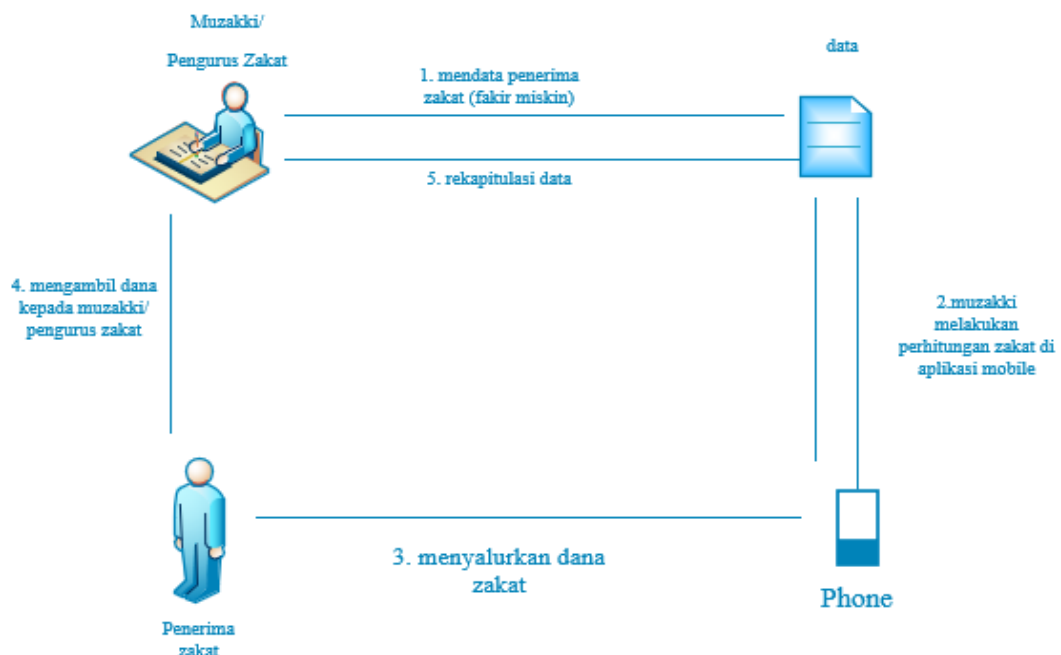
4.1.1.5 Analisa Sistem Usulan

Peneliti mengusulkan untuk membangun aplikasi perhitungan zakat dan fidyah, aplikasi yang akan dibangun diharapkan dapat membantu

masyarakat dalam menghitung jumlah zakat yang akan di keluarkan serta fidyah puasa. Dan memberikan informasi yang bermanfaat pada pengguna. Adapun analisa sistem usulan sebagai berikut :

1. Seorang muzakki atau pengurus lembaga zakat tidak perlu menghitung secara manual telah terintegrasi dengan *mobile*.
2. Aplikasi yang akan di bangun di lengkapi dengan perhitungan zakat ternak, zakat fitrah, zakat profesi dan zakat emas.
3. Aplikasi juga dapat mengkakulasi jumlah kaffarat yang harus kita bayar.
4. Aplikasi ini dilengkapi dengan panduan atau informasi mengenai zakat dan fidyah puasa, sehingga user mendapatkan wawasan dan pengentahuan tambahan dengan refrensi yang akurat mengenai informasi tersebut.

4.1.1.6 Gambar Analisa Sistem Usulan



Gambar 4.3 Analisa Sistem Berjalan

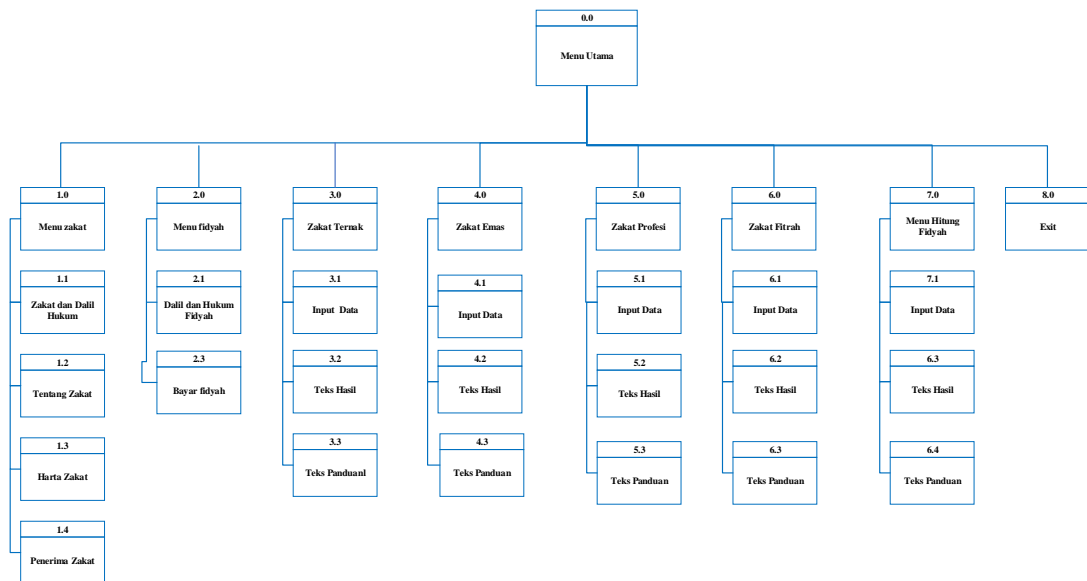
4.1.1.7 Narasi Sistem Usulan

Narasi sistem yang di usulkan, menceritakan alur yang berjalan antara user dan sistem (aplikasi zakat dan fidyah).

1. Seorang muzakki atau pengurus zakat mendata orang yang akan menerima zakat (secara manual terlebih dahulu).
2. muzakki atau pengurus zakat menginput data untuk perhitungan zakat pada aplikasi zakat yang di dukung dengan perhitungan fidyah puasa.
3. Narasi alur sistem aplikasi *mobile* user memulai (*start*) pada aplikasi zakat dan fidyah kemudian akan sistem akan menampilkan menu utama yang berisikan 8 item :
 - a. Menu info zakat (sub menu : zakat dan dalil hukum, penerima zakat, harta zakat, harta zakat, tentang zakat).
 - b. Menu fidyah menampilkan (sub menu: info fidyah, dalil dan hukum fidyah).
 - c. Menu hitung fidyah.
 - d. Menu zakat ternak
 - e. Menu zakat fitrah
 - f. Menu zakat emas
 - g. Menu zakat profesi
4. Sistem ini di dukung dengan informasi-informasi yang bermanfaat dan dapat menambah wawasan kepada *user* (pengguna) tentang zakat dan fidyah puasa.

4.1.2 Hipochart

Pada tahap ini peneliti menuangkan ide perancangan aplikasi yang akan dirancang dalam bentuk hipo . Dimana tahap ini menjelaskan diagram hirarki yang menggambarkan hubungan dari fungsi fungsi menu (yang menceritakan struktur paket). Keterangan masing masing fungsi digambarkan sebagai berikut:



Gambar 4.4 Hipochart aplikasi

4.1.2.1 Penjelasan hipochart aplikasi

Setelah hipochart selesai dirancang maka peneliti mendeskripsikan penjelasan Gambar 4.4 dalam bentuk *table*. Adapun *table* yang dihasilkan sebagai berikut :

Tabel 4.1 Penjelasan hipocahart

No	Nama	Penjelasan
1.0	Menu Zakat	Tampilan menu utama : menu zakat
1.1	Zakat dan Dalil Hukum	Informasi pembahasan zakat dan dalil hukum
1.2	Tentang Zakat	Informasi orang-orang yang menerima zakat
1.3	Harta Zakat	Informasi jenis harta yang di zakati
1.4	Tentang Zakat	Infromasi jenis-jenis zakat
2.0	Menu Fidyah	Tampilan menu utama : menu fidyah
2.1	Dalil dan Hukum Fidyah	Informasi mengenai dalil dan hukum fidyah
2.2	Penerima Fidyah	Infromasi orang-orang yang akan menerima fidyah
3.0	Zakat Ternak	Tampilan menu utama : Zakat Ternak

3.1	Input Data	Layout untuk menginput sejumlah data untuk di kalkulasi
3.2	Teks Hasil	Menampilkan teks box yang berisi hasil dari zakat ternak yang telah dikalkulasi diaplikasi <i>mobile</i>
3.3	Teks Panduan	Menampilkan teks box yang berisi tata cara perhitungan perhitungan zakat yang akan di hitung
4.0	Zakat Emas	Tampilan menu utama : Zakat Emas
4.1	Input Data	Layout untuk menginput sejumlah data untuk di kalkulasi
4.2	Teks Hasil	Menampilkan teks box yang berisi hasil dari zakat emas yang telah dikalkulasi diaplikasi <i>mobile</i>
4.3	Teks Panduan	Menampilkan teks box yang berisi tata cara perhitungan perhitungan zakat yang akan di hitung
5.0	Zakat Profesi	Tampilan menu utama : Zakat Profesi
5.1	Input Data	Layout untuk menginput sejumlah data untuk di kalkulasi
5.2	Teks Hasil	Menampilkan teks box yang berisi hasil dari zakat profesi yang telah dikalkulasi diaplikasi <i>mobile</i>
5.3	Teks Panduan	Menampilkan teks box yang berisi tata cara perhitungan perhitungan zakat yang akan di hitung
6.0	Zakat Fitrah	Tampilan menu utama : Zakat Fitrah
6.1	Input Data	Layout untuk menginput sejumlah data untuk di kalkulasi

6.2	Teks Hasil	Menampilkan teks box yang berisi hasil dari zakat fitrah yang telah dikalkulasi diaplikasi <i>mobile</i>
6.3	Teks Panduan	Menampilkan teks box yang berisi tata cara perhitungan perhitungan zakat fitrah yang akan di hitung
7.0	Hitung Fidyah	Tampilan menu utama : Zakat Fidyah
7.1	Input Data	Layout untuk menginput sejumlah data untuk di kalkulasi
7.2	Teks Hasil	Menampilkan teks box yang berisi hasil dari fidyah yang telah dikalkulasi diaplikasi <i>mobile</i>
8.0	Exit	Keluar dari laman

4.2 *Prototype*

Tahapan yang ke empat pada metode *Design Thinking* adalah *Prototype*, *prototype* dikenal sebagai rancangan awal suatu aplikasi atau produk yang akan dibuat. Peneliti telah memahami alur dari sistem yang akan dirancang, sehingga menjadi acuan yang menarik dalam merancang setiap fitur-fitur yang ada didalam aplikasi *mobile* ini. Dalam hal ini, peneliti mulai merancang beberapa *prototype*. Setelah mengetahui komponen sistem yang akan di rancang dengan *tools* yang sesuai untuk memenuhi kebutuhan *user*, maka dalam metode *design thinking* peneliti menggunakan usulan model *prototype* :

4.2.1 *Prototype Diagram*

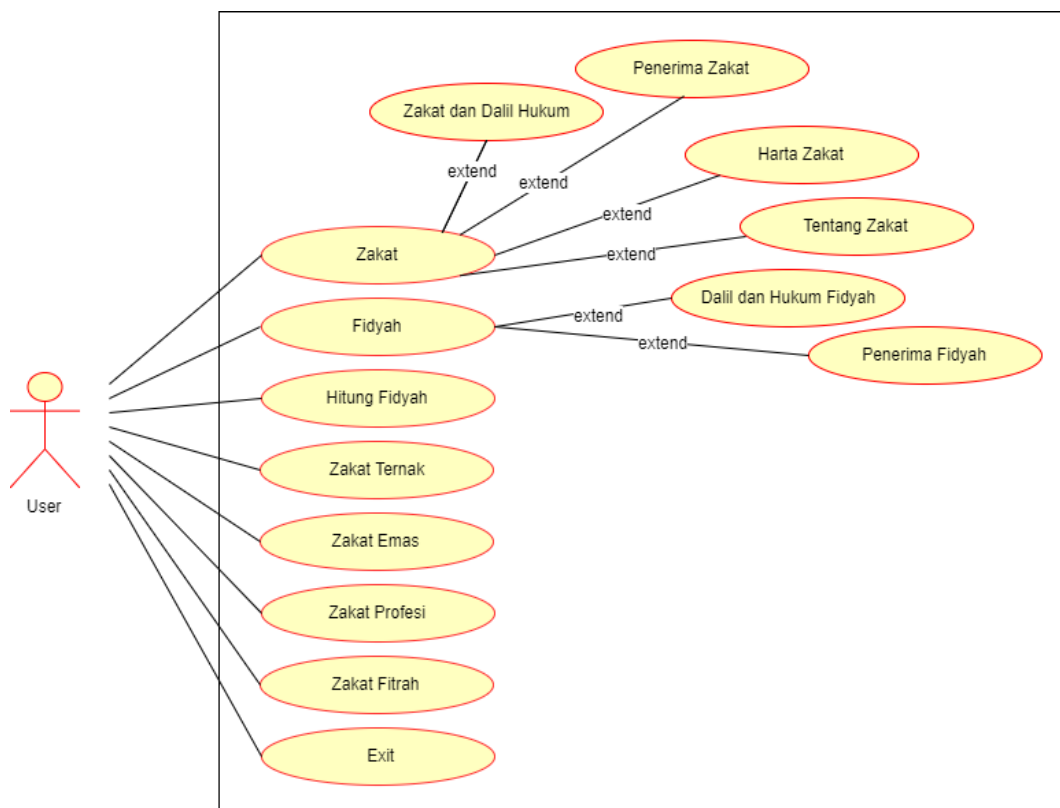
Peneliti membuat rancangan awal dengan penjabaran aplikasi yang akan di rancang dengan pemodelan UML (*Unified Modeling Language*). Terdiri dari *Use Case*, *Activity Diagram* dan *Sequence Diagram*.

1. *Use Case* : menceritakan hubungan *user* dan sistem.
2. *Activity Diagram* : menceritakan aktifitas yang di lakukan *user* dengan sistem.

3. *Sequence Diagram* menggambarkan interaksi antara sejumlah objek dalam urutan waktu.

4.2.1.1 Use Case

Dalam membangun sistem ini, peneliti menggunakan Use Case untuk membangun aplikasi yang akan di rancang, dimana use case berfungsi menggambarkan aplikasi secara *General* dan menggambar keseluruhan seperti gambar di bawah ini :



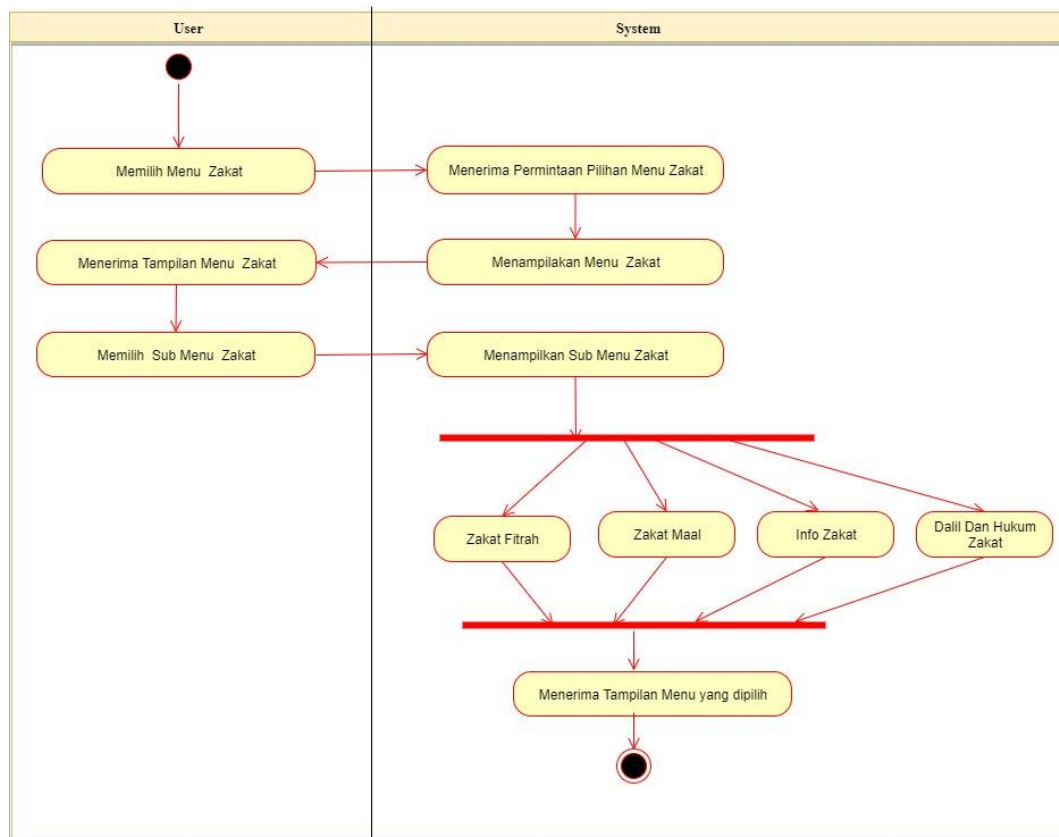
Gambar 4.5 Use Case Diagram Aplikasi

Dalam sistem ini peneliti membangun 8 menu utama yang tersedia pada dashboard yaitu :menu zakat, menu fidyah, hitung fidyah, zakat ternak, zakat emas, zakat profesi, zakat fitrah dan tombol exit. *User* dapat mengelola data ataupun informasi dari aplikasi zakat sesuai yang diinginkan.

4.2.1.2 Activity Diagram

Activity Diagram Menggambarkan rangkaian aliran dari aktivitas, di gunakan untuk mendeskripsikan aktifitas yang di bentuk dalam suatu operasi sehingga dapat juga di gunakan untuk aktifitas lainnya.

1. Activity Diagram Menu Zakat.

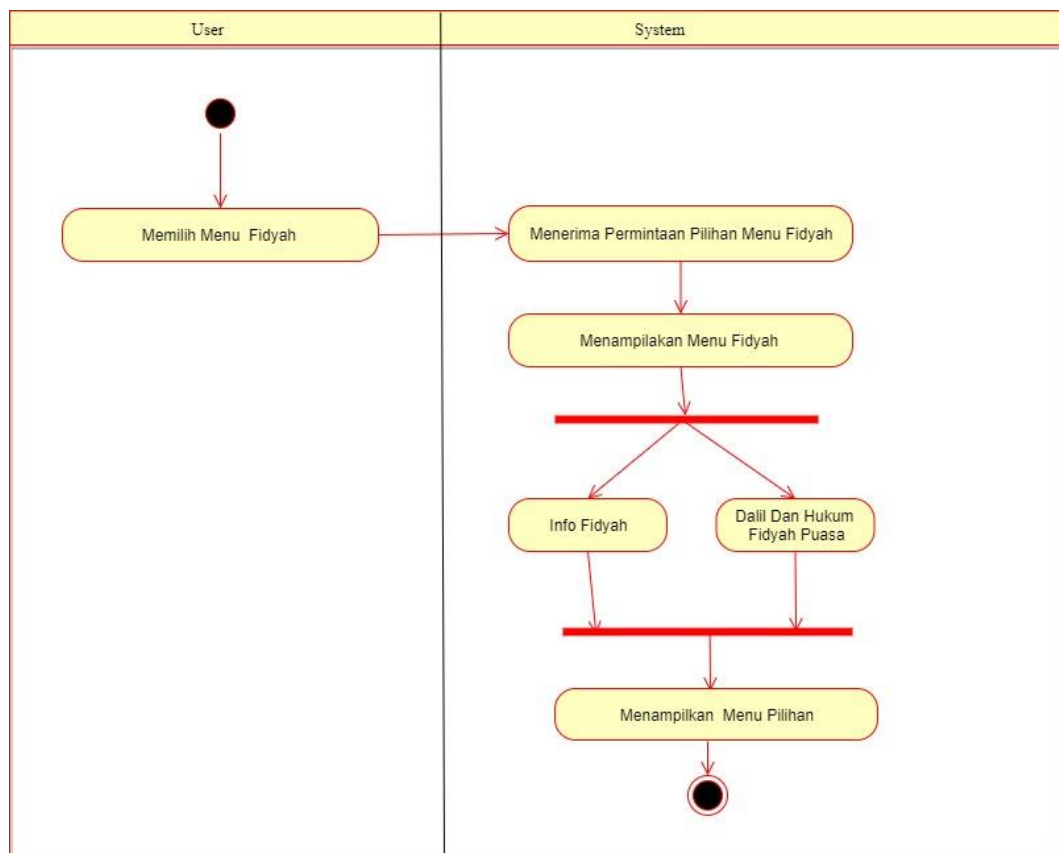


Gambar 4.6 Activity Diagram Menu Zakat

Activity diagram menu zakat menggambarkan alur aktifitas yang di lakukan oleh user , dimana user *start* aplikasi dan memilih menu zakat maka sistem akan menerima permintaan untuk menampilkan aplikasi yang di inginkan *user*, setelah user dapat memilih sub menu yang tersedia di

dashboard ada menu: zakat fitrah, zakat maal, info zakat serta dalil dan hukum zakat. Kemudian user akan menerima tampilan menu yang di pilih.

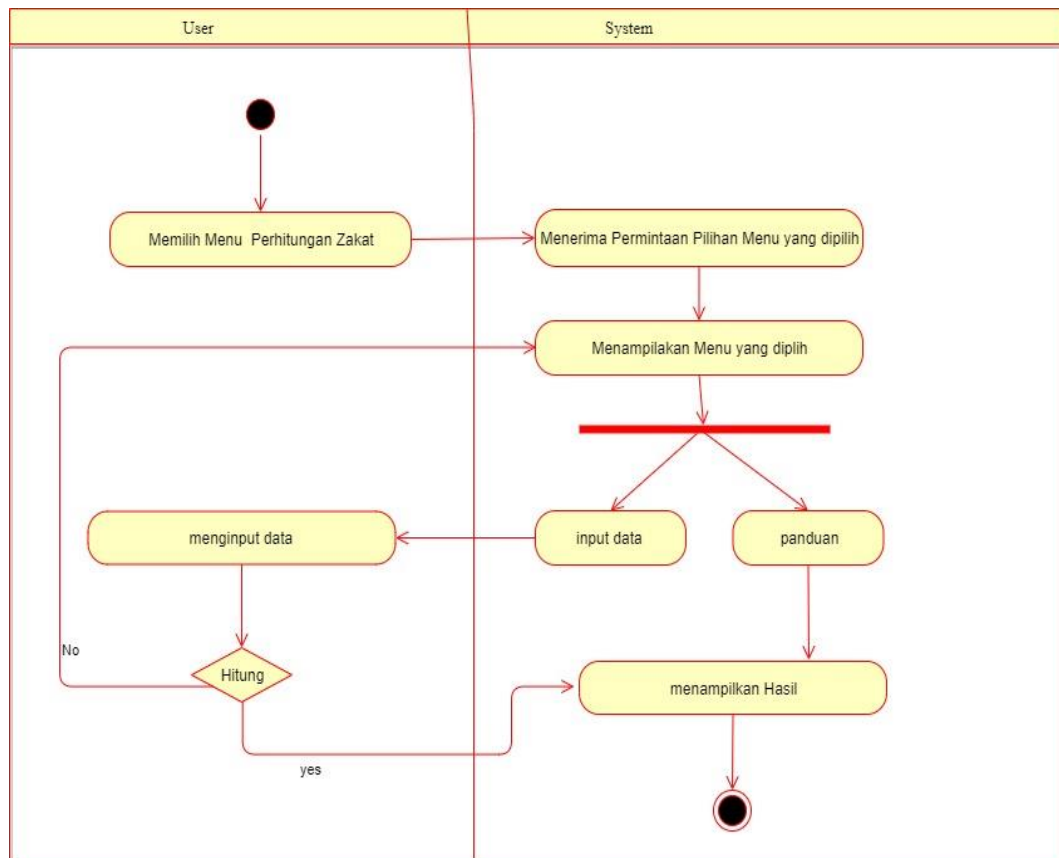
2. Activity Diagram Menu Fidyah



Gambar 4.7 Activity Diagram Menu Fidyah

Activity Diagram Menu Fidyah menggambarkan aktivitas user pada sistem yang sedang berjalan, dimana user *start* aplikasi kemudian memilih menu fidyah maka sistem akan menerima permintaan dan menampilkan sub menu yaitu : info fidyah serta dalil dan hukum fidyah puasa, kemudian *user* memilih salah satu menu yang tersedia di *dashboard* lalu sistem menerima permintaan untuk menampilkan menu yang dipilih, kemudian *user* menerima info dari menu yang telah di pilih.

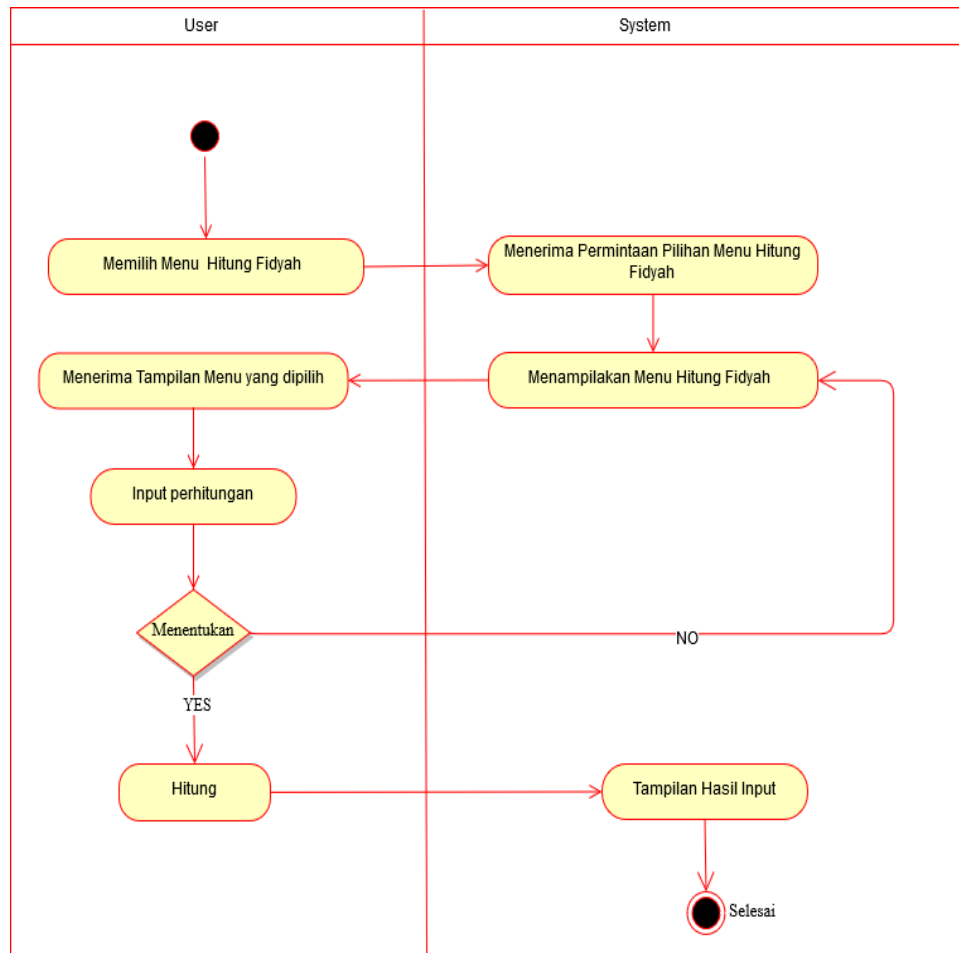
3. Activity Diagram Menu Hitung Zakat



Gambar 4.8 Activity Diagram Menu Hitung Zakat

Activity diagram menu hitung zakat menggambarkan aktivitas user pada sistem yang di rancang, dimana *user start* aplikasi kemudian memilih menu hitung zakat maka pada menu utama maka sistem menerima permintaan menu pilihan lalu sistem menampilkan menu yang dipilih. Kemudian user dua pilihan yaitu menginput data atau melihat informasi pada button “panduan” setelah menginput jumlah data maka user dapat menentukan hitungan jumlah zakat jika “Yes” maka sistem akan menampilkan hasil, jika memilih “No” maka sistem akan kembali menu yang dipilih.

4. Activity Diagram Menu Hitung Fidyah



Gambar 4.9 Activity Diagram Menu Hitung Fidyah

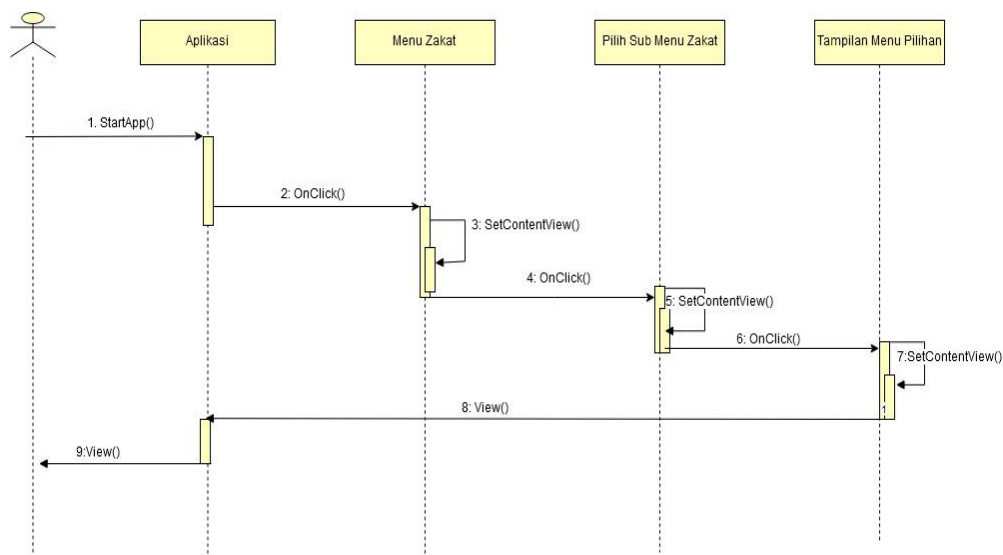
Activity menu hitung fidyah menggambarkan aktivitas user pada sistem yang sedang berjalan, dimana *user start* aplikasi kemudian memilih menu hitung fidyah maka sistem menerima permintaan menu pilihan lalu *user* dapat menginput data yang akan di kalkulasi oleh sistem. kemudian *user* dapat menentukan hitung jumlah fidyah yang akan dibayar jika “Yes” maka sistem akan menampilkan hasil outputnya, jika memilih “No” maka sistem akan kembali menampilkan menu hitung fidyah seperti awal lagi.

4.2.1.3 Sequence Diagram

Pada tahap ini *Sequence Diagram* menggambarkan interaksi antara sejumlah objek dalam urutan waktu pada aplikasi zakat dan fidyah. berikut ini adalah gambaran sequence untuk sistem yang di rancang.

1. Sequence Diagram Menu Zakat

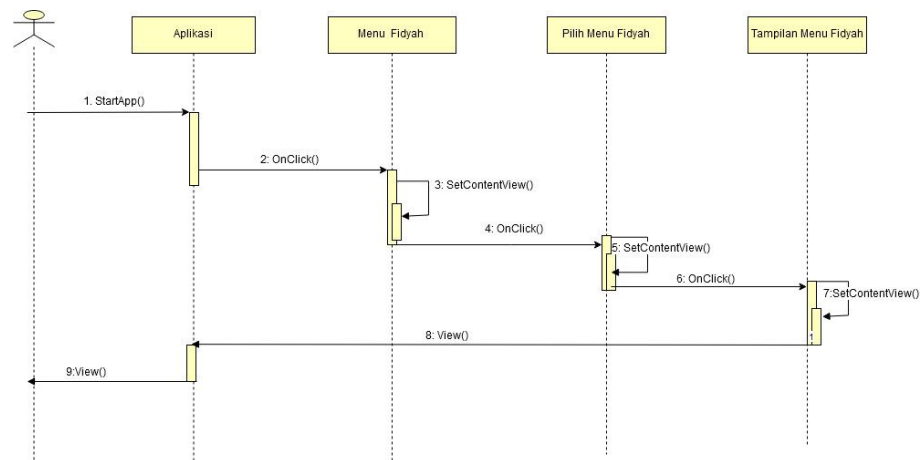
Untuk menampilkan menu yang diinginkan user diwajibkan klik pada menu yang tersedia pada dashboard, dimana setelah klik pada menu zakat, user dapat memilih sub menu yang di inginkan seperti : zakat fitrah, zakat maal, info zakat dan dalil dan hukum zakat. Setelah itu user dapat membaca info atau pengetahuan mengenai zakat.



Gamabar 4.10 *Sequence Diagram* Menu Zakat

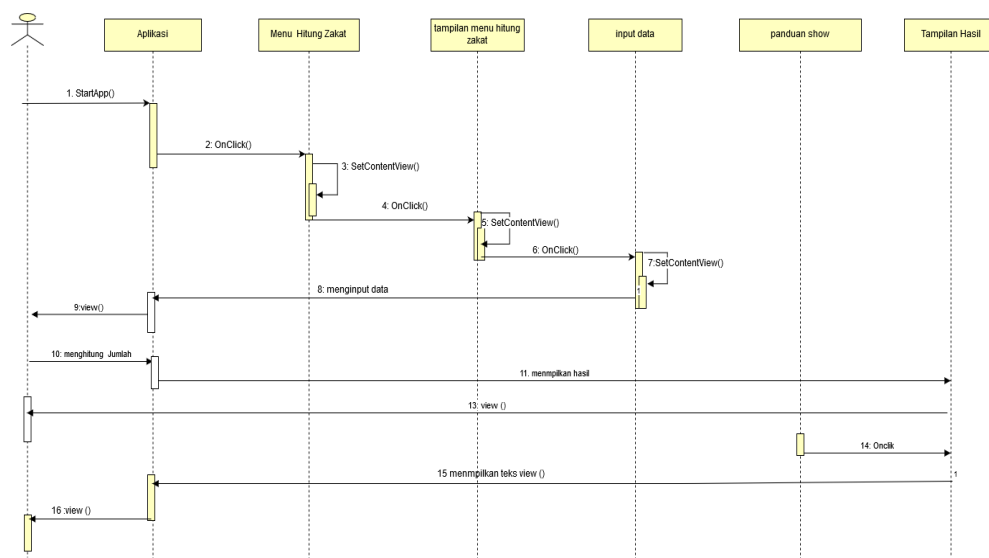
2. Sequence Diagram Menu Fidyah

Pada menu ini user dapat melihat info mengenai fidyah puasa dengan cara *start* aplikasi zakat dan klik menu Fidyah, dalam menu fidyah tersedia 2 sub menu yaitu : info fidyah, dalil dan hukum fidyah puasa. Setelah user memilih salah satu dari menu tersebut maka sistem menerima permintaan untuk menampilkan menu yang di pilih.

Gambar 4.11 *Sequence Diagram* Menu Fidyah

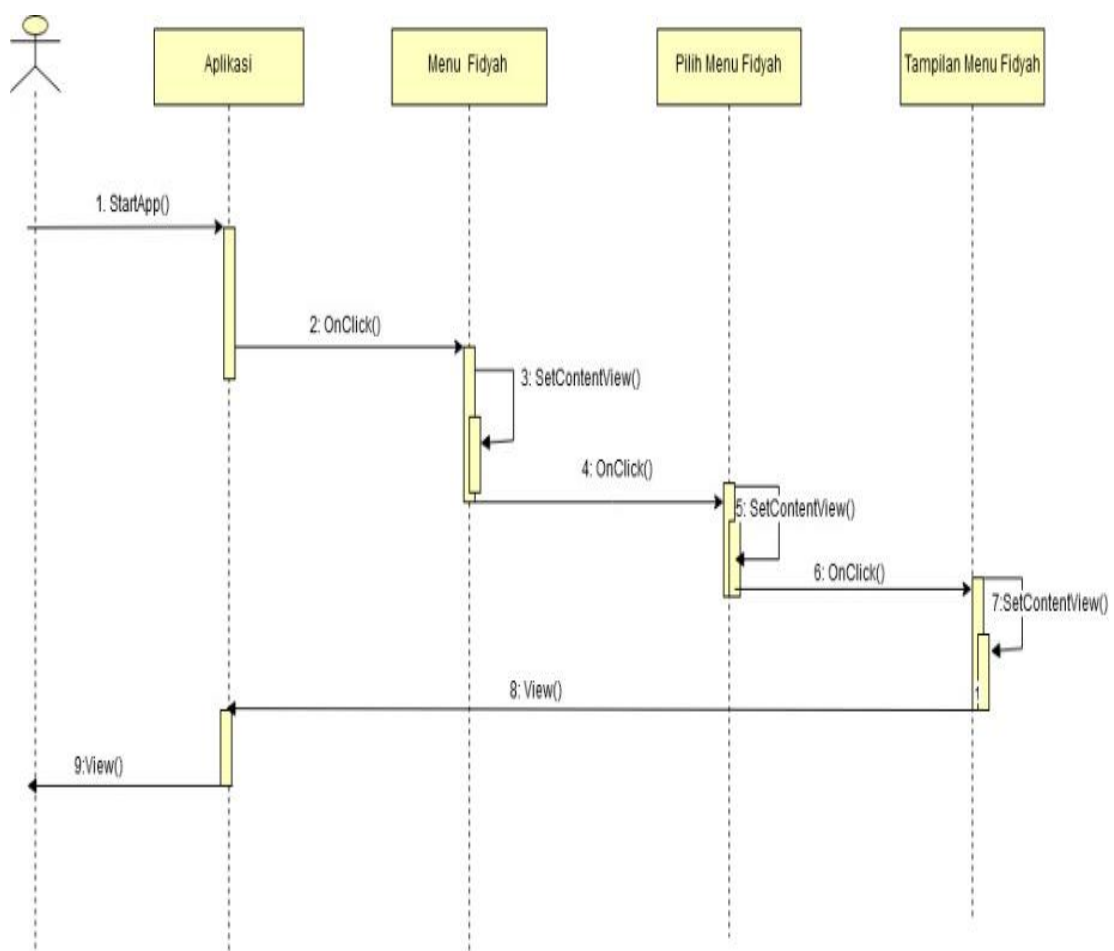
3. *Sequence Diagram* Menu Hitung Zakat

Untuk melihat hasil dari suatu zakat yang akan di keluarkan oleh muzakki, maka Squence Diagram menggambarkan Menu Hitung Zakat setiap alur dari sistem yang di gunakan oleh user seperti gambar 3.9. dalam menu ini user melakukan penginputan data untuk mengetahui nominal yang akan di bayar oleh muzakki sesuai dengan jenis zakat yang akan di kalkulasi pada aplikasi *mobile*.

Gamabar 4.12 *Sequence Diagram* Menu Hitung Zakat

4. *Sequence Diagram* Menu Hitung Fidyah

Pada menu ini user dapat mengetahui jumlah fidyah yang akan di bayar karena meninggalkan puasa sesuai dengan jumlah puasa yang di tinggalkan. User membuka aplikasi dan memilih (klik) menu fidyah dan melakukan inputan data.

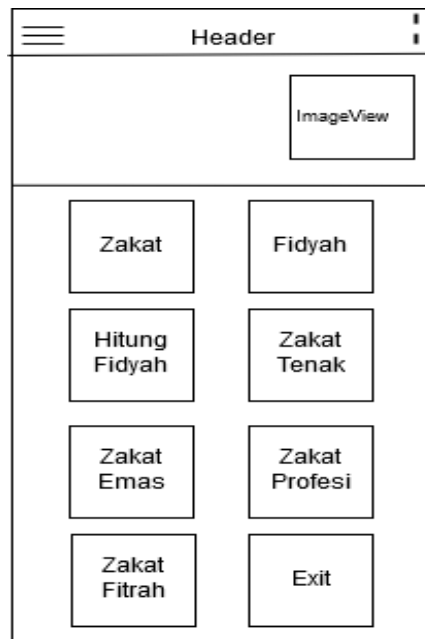


Gambar 4.13 *Sequence Diagram* Menu Hitung Fidyah

4.2.2 *Prototype Interface*

Peneliti merancang *interface* aplikasi yang di rancang. Adapun rancangan aplikasi sebagai berikut :

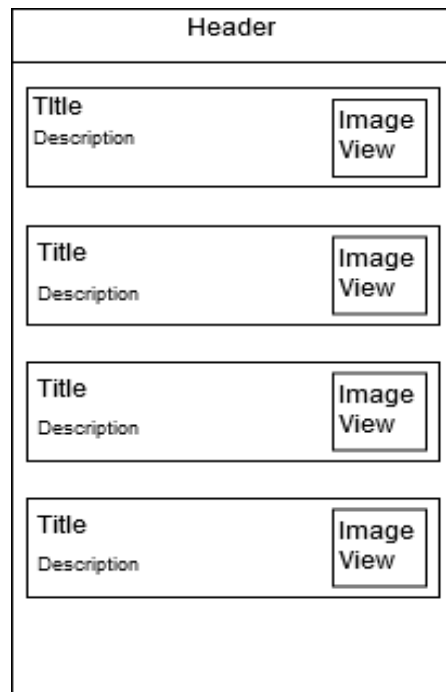
1. Menu Utama



Gambar 4.14 Dashboard (menu utama)

Pada menu utama menampilkan 6 menu di antaranya ada menu zakat, menu fidyah, menu hitung fidyah, menu zakat ternak, menu zakat emas, menu zakat profesi, menu zakat fitrah dan tombol exit.

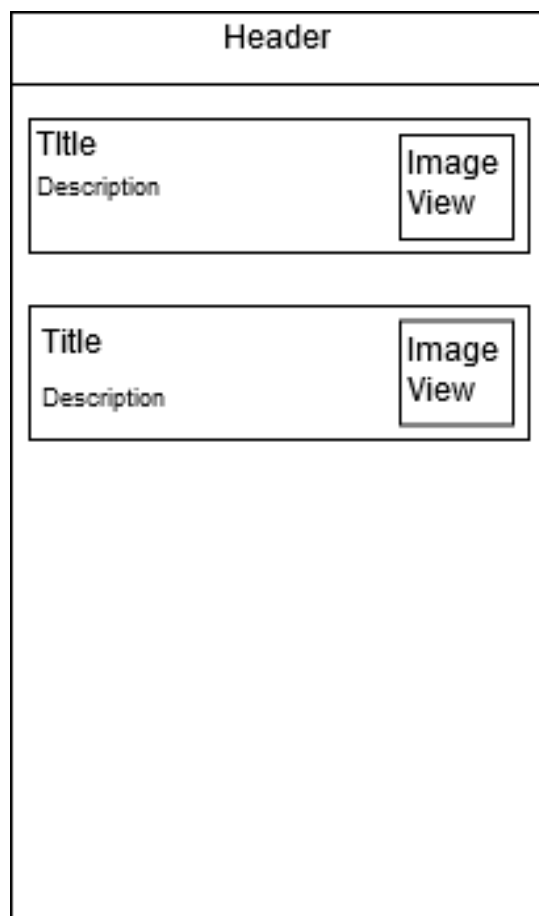
2. Menu Zakat



Gambar 4.15 Sub Menu dari Menu Zakat

Pada menu zakat terdapat 4 sub menu yang akan di rancang, dimana textview Title berisi judul besar, dan textview Description menampilkan penjelasan singkat dari textview title. Serta imageview adalah hiasan icon/gambar yang akan membuat tampilan menjadi menarik.

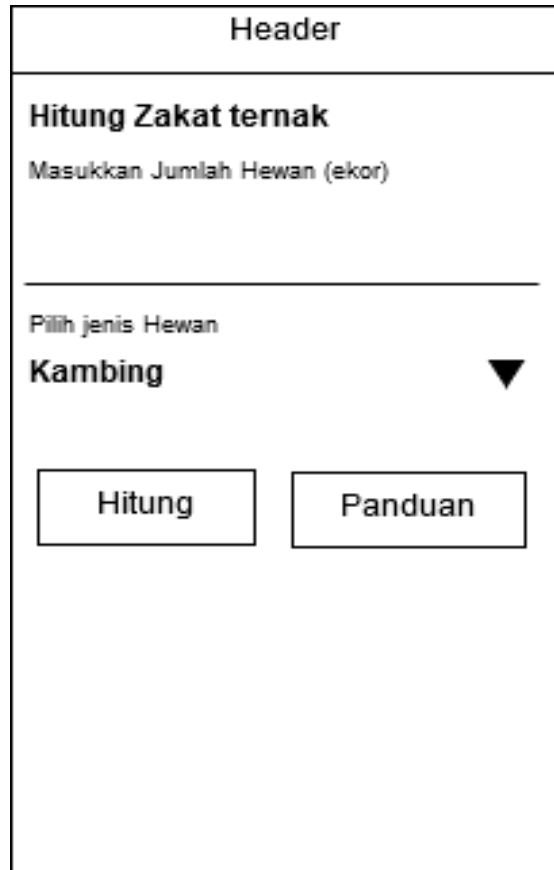
3. Menu Fidyah



Gambar 4.16 Sub Menu Dati Menu Fidyah

Pada menu Fidyah terdapat 2 sub menu yang akan di rancang, dimana textview Title berisi judul besar, dan textview Description menampilkan penjelasan singkat dari textview title. Serta imageview adalah hiasan icon/gambar yang akan membuat tampilan menjadi menarik.

4. Menu Zakat Ternak



Header

Hitung Zakat ternak

Masukkan Jumlah Hewan (ekor)

Pilih jenis Hewan

Kambing ▼

Hitung Panduan

Gambar 4.17 Menu Hitung Zakat Ternak

Pada menu hitung zakat menampilkan 3 buah textview, 1 edittext, 1 spinner untuk memilih jenis hewan yang akan dipanggil, serta 2 buah button yaitu button hitung untuk menampilkan hasil dari perhitungan dan button panduan untuk menginformasikan informasi mengenai zakat ternak.

5. Menu Zakat Emas

Header	
Hitung Zakat Emas	
Jumlah Emas yang Dimiliki (gram)	
<hr/>	
Harga Emas Saat Itu	
<hr/>	
Hitung	Panduan

Gambar 4.18 Menu Hitung Zakat Emas

Pada menu hitung zakat emas terdapat 3 buah textview, 2 buah edittext, serta 2 buah button yaitu button hitung untuk menampilkan hasil dari perhitungan yang telah terintegritas diaplikasi *mobile* dan button panduan untuk menginformasikan tatacara perhitungan zakat emas yang dilengkapi dengan contoh untuk menambah wawasan *user*.

6. Menu Zakat Profesi

Header	
Hitung Zakat Profesi	
Penghasilan Keseluruhan Perbulan	
<hr/>	
Pendapatan Bonus dan Lainnya Perbulan	
<hr/>	
Pengeluaran Kebutuhan Pokok Perbulan	
<hr/>	
Harga Beras Saat In (Per Kg)	
<hr/>	
Hitung	Panduan

Gambar 4.19 Menu Hitung Zakat Profesi

Pada menu hitung zakat profesi terdapat 5 buah textview, 4 buah edittext, serta 2 buah button yaitu button hitung untuk menampilkan hasil dari perhitungan yang telah terintegritas diaplikasi *mobile* dan button panduan untuk menginformasikan tatacara perhitungan zakat profesi yang di lengkapi dengan contoh perhitungan secara manual.

7. Menu Zakat Fitrah

Header	
Hitung Zakat Fitrah	
Harga Beras saat ini (per Kg)	
<hr/>	
Zakat Fitrah per Jiwa (2,5 Kg)	
<hr/>	
Total Jiwa yang Membayar zakat Fitrah	
<hr/>	
Hitung	Panduan

Gambar 4.20 Menu Hitung Zakat Fitrah

Pada menu hitung zakat emas terdapat 4 buah textview, 3 buah edittext, serta 2 buah button yaitu button hitung untuk menampilkan hasil dari perhitungan yang telah terintegritas diaplikasi *mobile* dan button panduan untuk menginformasikan tatacara perhitungan zakat fitrah.

4.3 Test (pengujian)

Tahap yang kelima dalam metode *design thinking* adalah pengujian aplikasi, pengujian aplikasi ini di lakukan dengan menggunakan metode *Blackbox testing* untuk melihat apakah aplikasi ini, berjalan sesuai dengan kebutuhan, dan melihat apakah aplikasi ini sudah layak digunakan oleh masyarakat luas

4.3.1 Menentukan Hak Akses

Hak akses ini digunakan oleh orang muslim yang ada di Indonesia, karena aplikasi ini bersifat umum dengan cara melakukan instalasi aplikasi di *play store* pada *smartphone* android pengguna.

4.3.2 Black Box Testing

Pengujian sistem yang digunakan menggunakan metode blackbox testing. Metode ini merupakan pengujian yang memfokuskan pada keperluan fungsional dari *software*. Karena itu pengujian blackbox memungkinkan pengembangan *software* untuk membuat himpunan kondisi input yang akan melatih seluruh syarat-syarat fungsional suatu program.

Tabel 4.2 *Black Box Testing*

No	Aktivitas Pengujian	Tampilan	Pengujian	Fungsi	Tidak Fungsi	Penilaian
1	Masuk Halaman	Menampilkan 8 menu	Scroll View Action Bar Drawel Navigation Bar	✓ ✓ ✓ ✓	- - - -	100 %
2	Klik Menu Info Zakat	4 Menu (ListView)	Scroll View Teks View Toolbar Back	✓ ✓ ✓	- - -	100 %
3	Klik Menu Info Fidyah	Menampilkan 2 menu	Scroll View Teks View Toolbar Back	✓ ✓ ✓	- - -	100%
4	Klik Menu Zakat Ternak	Menampilkan Layout untuk menginput data	Spinner (pilih jenis hewan). Edit Text (input data) Teks Hasil Teks Panduan	✓ ✓ ✓ ✓	- - - -	100 %

			Toolbar Back		-	
5	Klik Menu Hitung Fidyah	Menampilkan Layout untuk menginput data	Edit Text (input data) Teks Hasil	✓ ✓	- -	100%
6	Klik Menu Zakat Emas	Menampilkan Layout untuk menginput data	Edit Text (input data) Teks Hasil Teks Panduan Toolbar Back	✓ ✓ ✓ ✓	- - - -	100%
7	Klik Menu Zakat Profesi	Menampilkan Layout untuk menginput data	Edit Text (input data) Teks Hasil Teks Panduan Toolbar Back	✓ ✓ ✓ ✓	- - - -	100%
8	Klik Menu Zakat Fitrah	Menampilkan Layout untuk menginput data	Edit Text (input data) Teks Hasil Teks Panduan Toolbar Back	✓ ✓ ✓ ✓	- - - -	100%
9	Klik exit	Logout	Logout dari aplikasi	✓	-	100%

Tabel 4.3 pengujian fungsi

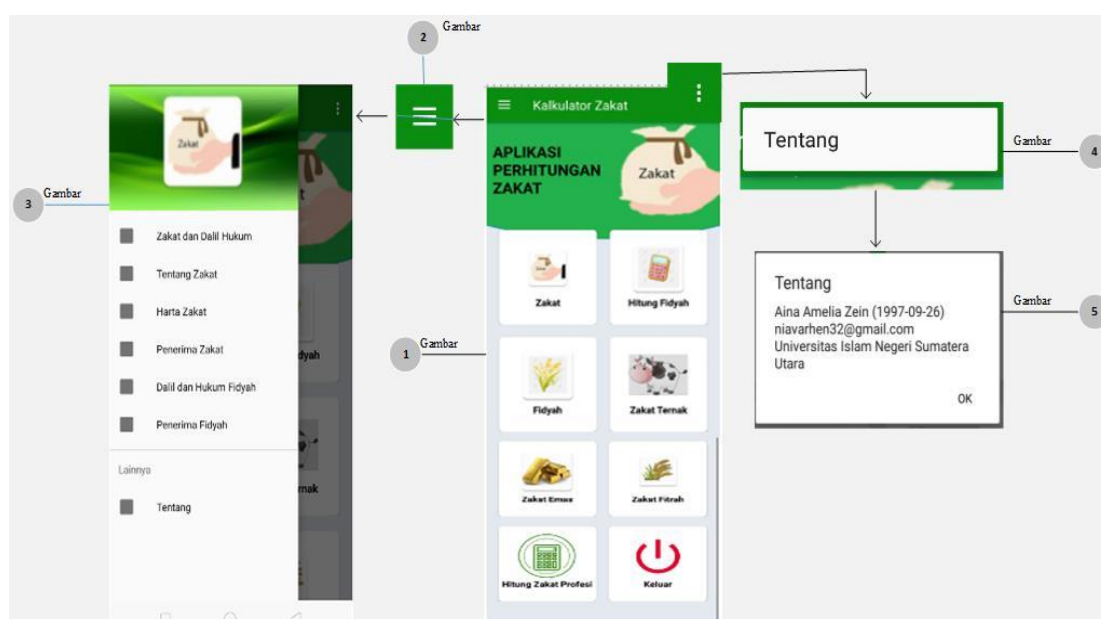
No	Nama	Tampilan	Input		Output		Panduan	Hasil Test
			berhasil	tidak	berhasil	tidak		
1	menu zakat	berhasil	✓	-	✓	-	✓	berhasil

2	fidyah	berhasil	✓	-	✓	-	✓	berhasil
3	zakat ternak	berhasil	✓	-	✓	-	✓	berhasil
4	hitung fidyah	berhasil	✓	-	✓	-	✓	berhasil
5	zakat emas	berhasil	✓	-	✓	-	✓	berhasil
6	zakat fitrah	berhasil	✓	-	✓	-	✓	berhasil
7	zakat profesi	berhasil	✓	-	✓	-	✓	berhasil

4.3.3 Implementasi

pada tahap ini, peneliti mengimplentasikan aplikasi yang telah jadi beserta penjelasan mengenai dari fungsi-fungsi yang terletak pada aplikasi zakat.

1. Menu Utama



Gambar 4.21 Tampilan Utama Aplikasi

Tabel 4.4. Pejelasan Menu Utama

Gambar	Nama	Fungsi
1	Menu Utama	Pada menu utama implemtasi sistem menampilkan 8 item yang dapat dilihat pada Gambar.20 diantaranya adalah zakat, hitung fidyah, fidyah, zakat ternak, zakat emas, zakat fitrah, zakat profesi
2	Action Bar Group	Jika user mengklik gambar 2 maka sistem akan menampilkan 7 listview di antaranya adalah : Zakat dan dalil hukum, Tentang zakat, Harta Zakat, Penerima zakat, Dalil dan hukum fidyah, penerima fidyah, tentang.
3	Listview ActionBar	Adapaun menu dari di antaranya adalah : Zakat dan dalil hukum, Tentang zakat, Harta Zakat, Penerima zakat, Dalil dan hukum fidyah, penerima fidyah, tentang.
4	Item Group	3 titik sebelah kanan atas berfungsi menampilkan menu tentang
5	Menu tentang	Jika <i>user</i> klik menu “tentang” maka sistem akan menampilkan teksbox yang berupa informasi untuk mengkontak admin.

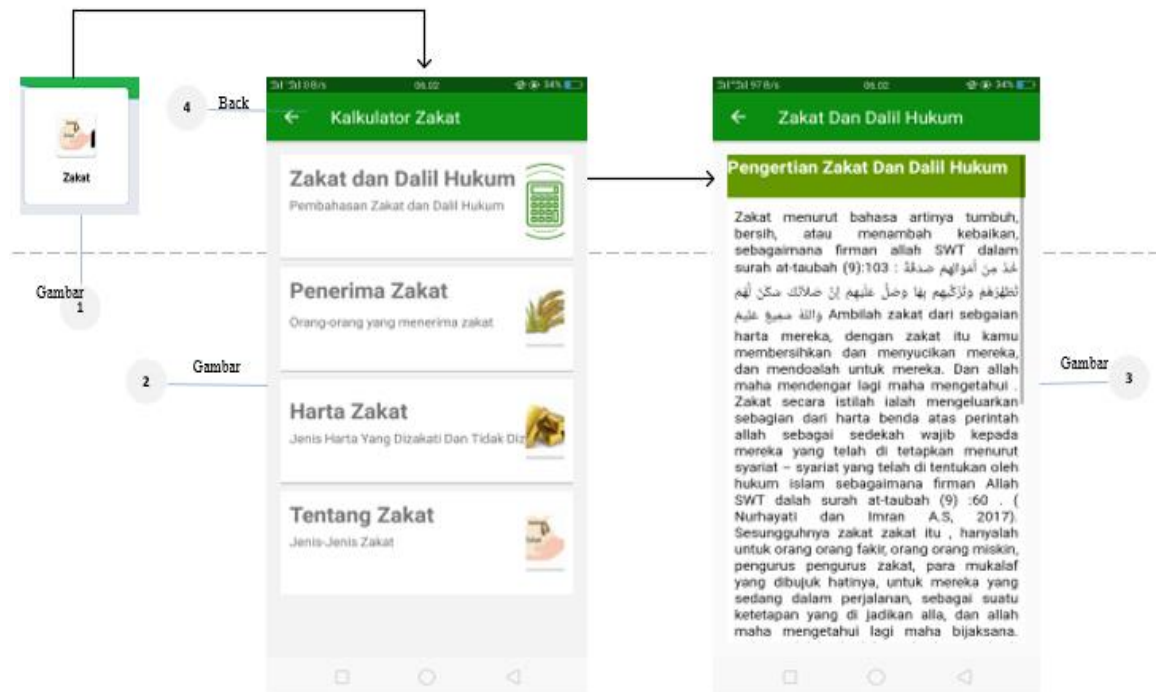
2. Menu Utama (drawel)



Gambar 4.22 Tampilan Drawel

Pada menu ini, actionbar menampilkan informasi yang bermanfaat bagi pengguna, dimana dari setiap menu listview yang terletak pada navigationbar berfungsi menampilkan informasi jika pengguna mengklik menu yang tersedia.

3. Menu Zakat

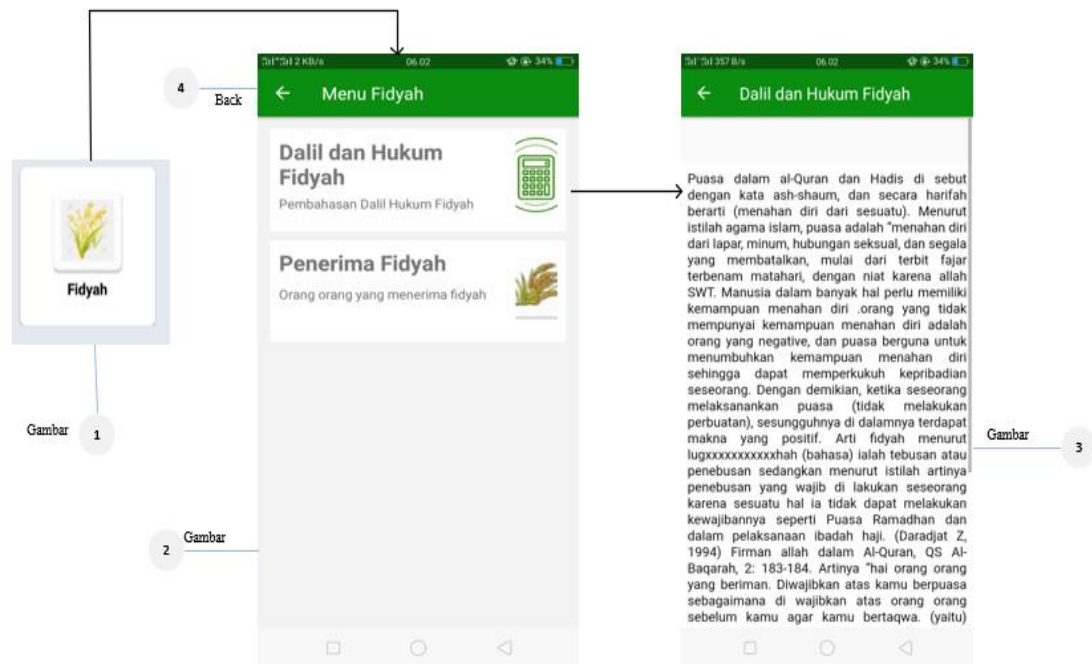


Gambar 4.23 Sub menu dari menu zakat

Tabel 4.5 Sub menu dari menu zakat

Gambar	Nama	Penjelasan
1	Icon menu zakat	Tempilan dari menu utama
2	Sub menu zakat	Jika <i>user</i> klik menu zakat maka sistem akan menampilkan listview yang terdiri dari 4 menu
3	Infomasi	Jika user klik salah satu listview pada menu yang tersedia pada aplikasi maka sistem akan menampilkan informasi-informasi yang bermanfaat.
4	Panah	Jika user klik tanda panah pada kiri atas maka sistem akan kembali pada laman menu yang dibuka sebelumnya.

4. Menu fidyah

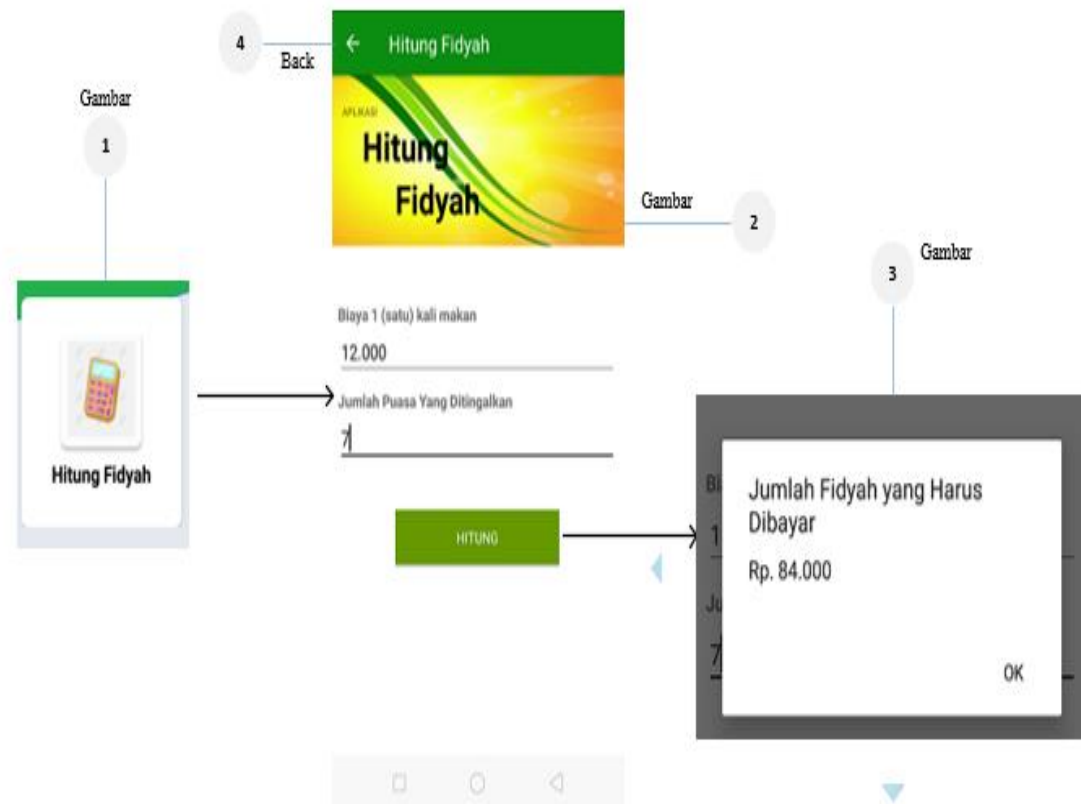


Gambar 4.24 Sub menu dari menu fidyah

Tabel 4.6 sub menu dari menu fidyah

Gambar	Nama	Penjelasan
1	Icon menu fidyah	Tempilan dari menu utama
2	Sub menu fidyah	Jika <i>user</i> klik menu fidyah maka sistem akan menampilkan listview yang terdiri dari 2 menu
3	Infomasi	Jika user klik salah satu listview pada menu yang tersedia pada aplikasi maka sistem akan menampilkan informasi-informasi yang bermanfaat.
4	Panah	Jika user klik tanda panah pada kiri atas maka sistem akan kembali pada laman menu yang dibuka sebelumnya.

5. Menu Hitung fidyah



Gambar 4.25 Tampilan Menu Hitung fidyah

Tabel 4.7 Menu Perhitungan Fidyah

Gambar	Nama	Penjelasan
1	Icon hitung fidyah	Tempilan dari menu utama
2	Layout menu fidyah	Jika <i>user</i> klik menu hitung fidyah maka sistem akan menampilkan layout hitung fidyah dimana pada layout ini, <i>user</i> melakukan <u>penginputan data sesuai</u> yang tertera pada aplikasi
3	Textbox info	Jika user telah menginput sejumlah data maka user akan klik button hitung yang akan menghasilkan textbox perintah jumlah fidyah yang akan di bayar.

4	Panah Back	Jika user klik tanda panah pada kiri atas maka sistem akan kembali pada laman menu yang dibuka sebelumnya.
---	------------	--

6. Menu Zakat ternak



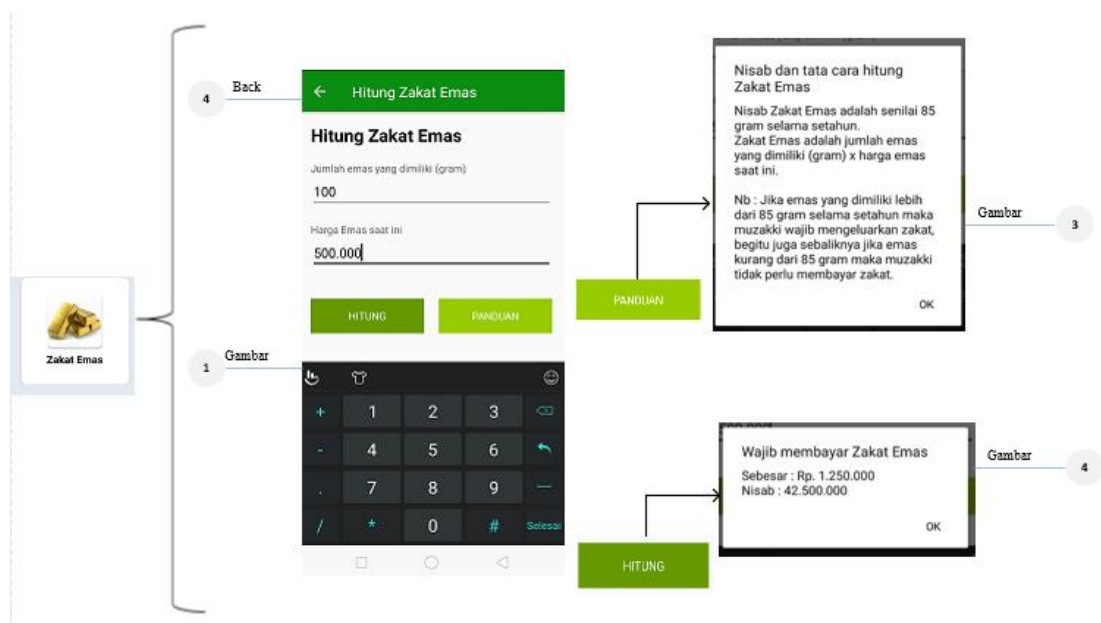
Gambar 4.26 Tampilan menu Zakat ternak

Tabel 4.8 penjelasan zakat ternak

Gambar	Nama	Penjelasan
1	Tampilan Layout zakat ternak	Pada layout ini user dapat mengetahui jumlah zakat ternak yang akan di keluarkan zakat nya dengan memasukkan jumlah hewan dan pilih jenis hewan
2	Drop Down	Pada gambar 2 terdapat drop down yang berfungsi agar user dapat memilih jenis hewan dari zakat ternak.

3	Button hitung	Jika user telah menginput jumlah dan jenis hewan maka button hitung akan menghasilkan textbox yang berisi informasi jumlah zakat yang wajib di keluarkan.
4	Button panduan	Jika user klik button panduan maka sistem akan menampilkan informasi mengenai zakat ternak.

7. Menu Zakat Emas



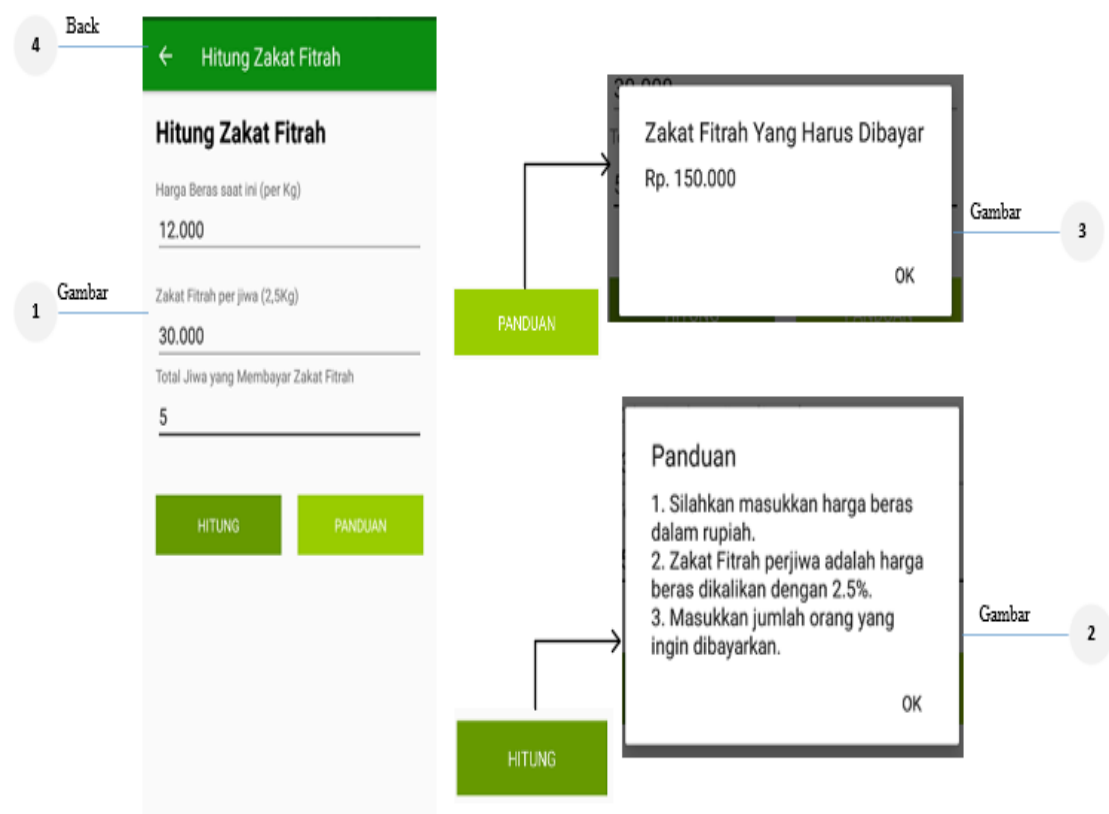
Gambar 4.27 Tampilan menu Zakat Emas

Tabel 4.9 penjelasan zakat emas

Gambar	Nama	Penjelasan
1	Tampilan Layout zakat emas	Pada layout ini user dapat menginput jumlah emas yang dimiliki serta harga emas saat ini
2	Button hitung	Jika user telah menginput sejumlah perintah pada layout zakat emas, maka user akan menekan button hitung mengetahui jumlah zakat yang wajib

		di bayar.dan button hitung akan menghasilkan textbox informasi
3	Button panduan	Jika user klik button panduan maka sistem akan menampilkan informasi mengenai zakat emas.
4	Panah Back	Jika user klik tanda panah pada kiri atas maka sistem akan kembali pada laman menu yang dibuka sebelumnya.

8. Menu Zakat Fitrah



Gambar 4.28 Tampilan menu Zakat Fitrah

Tabel 4.10 penjelasan zakat fitrah

Gambar	Nama	Penjelasan
1	Tampilan Layout zakat fitrah	Pada layout ini user dapat menginput harga beras, yang terdeteksi secara otomatis, dan mnginput jumlah jiwa yang akan membayar zakat
2	Button hitung	Jika user telah menginput sejumlah perintah pada layout zakat fitrah, maka user akan menekan button hitung mengetahui jumlah zakat yang wajib di bayar.dan button hitung akan menghasilkan textbox informasi
3	Button panduan	Jika user klik button panduan maka sistem akan menmpilkan informasi mengenai zakat fitrah.
4	Panah Back	Jika user klik tanda panah pada kiri atas maka sistem akan kembali pada laman menu yang dibuka sebelumnya.

9. Menu Zakat Profesi



Gambar 4.29 Tampilan Zakat Profesi

Tabel 4.11 penjelasan zakat profesi

Gambar	Nama	Penjelasan
1	Tampilan Layout zakat profesi	Pada layout ini user dapat menginput penghasilan, bonus dan pendapatan lainnya perbulan, menginput pengeluaran selama sebulan, serta harga beras saat ini (kg).
2	Button hitung	Jika user telah menginput sejumlah perintah pada layout zakat profesi, maka user akan menekan button hitung untuk mengetahui jumlah zakat yang akan dibayar.dan button hitung akan menghasilkan textbox yang berisi informasi
3	Button panduan	Jika user klik button panduan maka sistem akan menampilkan informasi mengenai zakat profesi dan tata cara perhitungannya.
4	Panah Back	Jika user klik tanda panah pada kiri atas maka sistem akan kembali pada laman menu yang dibuka sebelumnya.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan pembahasan pada bab-bab sebelumnya, maka dapat ditarik kesimpulan sebagai berikut:

1. Dengan adanya aplikasi ini, seorang muzakki dapat mengetahui jumlah zakat yang akan dibayar secara akurat karna telah terintegrasi dalam aplikasi *mobile*.
2. Dengan adanya sistem ini menambah pengetahuan seseorang atau pengguna (*user*) yang didukung dengan informasi tentang zakat dan fidyah puasa.
3. Dengan adanya sistem ini pengguna juga mendapatkan informasi tata cara perhitungan zakat secara manual.
4. Aplikasi ini menyediakan fitur *contact me* yang berhubungan dengan developer untuk mengantisipasi terjadi kesalahan dalam perhitungan.

5.2 Saran

Berdasarkan simpulan dan analisis yang telah dilakukan, maka terdapat beberapa saran sebagai berikut:

1. Melakukan kerja sama kepada pihak badan zakat agar membayar zakat lebih mudah tanpa harus datang kelokasi.
2. Menambahkan database untuk kariawan/pekerja dalam suatu instansi atau perusahaan sehingga gaji perbulan yang dihasilkan terpotong secara otomatis untuk pembayar zakat profesi.
3. Perlu adanya pengembangan aplikasi sehingga aplikasi ini multi platfrom.

DAFTAR PUSTAKA

- Atunnisa, R., Satria, E., & Cahyana, R. (2014). Pengembangan Aplikasi Zakat Berbasis Android Menggunakan Metode Prototype. *Algoritma*.
- hussein, A. s. (2018). *metode design thinking untuk inovasi bisnis*. Malang : UB Press.
- Daradjat Z., 1994, *Dasar-Dasar Agama Islam*, Penerbit Pt Bulan Bintang, Jakarta.
- DR .Abdullah Bin Muhammad Bin Ahmad Ath-Thayyar, 2010 "*Fikih Ibadah*". Penerbit Media Zikir, Surakarta-Jawah Tengah.
- Komputer, w. (2013). *Step by Step Menjadi Programmer Android*, ANDI OFFSET, Semarang.
- komputer, w. (2012). *Membuat Aplikasi Android untuk Tablet dan Handphone*. Jakarta: Pt Elex media komputindo.
- komputer, w. (2014). *Membangun Aplikasi Mobile Cross Platform Dengan Phonegap*. Jakarta : PT Elex Media Komputindo.
- M, S., & M, S. (n.d.). *Rancangan Aplikasi Menghitung Zakat*. 9(3), 290–300.
- Nurhayati, dan Imran, A.S., 2017, *Fidh dan Ushul Fiqh*, Prenadamedia Group, Depok.
- Nurhayati dan Imran A, S;. (2017). *Fidh dan Ushul Fiqh*. Depok: Prenadamedia Group.
- Razi, A. A., Mutiaz, I. R., & Setiawan, P. (2018). Penerapan Metode Design Thinking Pada Model Perancangan Ui/Ux Aplikasi Penanganan Laporan Kehilangan Dan Temuan Barang Tercecer. *Desain Komunikasi Visual, Manajemen Desain Dan Periklanan (Demandia)*, 3(02), 75. <https://doi.org/10.25124/demandia.v3i02.1549>
- Rosa A.S, M. (2016). *Rekayasa Perangkat Lunak*. Bandung: informatika.

- Rian, H. (2014). Perancangan Sistem Informasi Jasa Servis Pada Astra Honda Gandaria Jakarta. *Jurnal Lentera Ict*.
- Rickyanto, I., 2003, *Dasar Pemrograman Berorientasi Objek dengan Java 2(JDK 1.4)*, ANDI, Yogyakarta.
- Rochim, A., 1436 H, *Buku Panduan Zakat*, Yayasan Domoet Dwafa Republika, Medan.
- Rusyd, I., 2010, *Bidayatul Mujtahid wa Nihayatul Muqtashid*, Makatabah Asy-Syuruq Ad-Dauliyyah, Jakarta Timur.
- Satyaputra, A., dan Aritonang, E.M., 2014, *Java for Beginners with Eclipse 4.2 Juno*, Penerbit PT Elex Media Komputindo, Jakarta.
- Sibarani, N. S., Munawar, G., & Wisnuadhi, B. (2018). *Analisis Performa Aplikasi Android Pada Bahasa Pemrograman Java dan Analisis Performa Aplikasi Android Pada Bahasa Pemrograman Java dan Kotlin*. (December).
- Supardi, Y., 2014, *Semua Biasa Menjadi Programmer Android*, PT Elex Media Komputindo. Jakarta.
- Supardi, I. Y. (2014). *semua bisa menjadi programmer android*. jakarta: PT Elex Media Komputindo.
- Suendri. (2018). Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem (Studi Kasus : UIN Sumatera Utara Medan). *Jurnal Ilmu Komputer Dan Informatika*, 3(1), 1–9.
- P.Juhara, Z. (2016). *Panduan Lengkap Pemrograman Android*. Yogyakarta: ANDI.
- Pertiwi, J. Z. (2016). *Panduan Lengkap Pemrograman Android*. Yogyakarta: ANDI.
- wadi, h. (2018). *pemrograman android untuk pelajar dan mahasiswa*. semarang: andi.

yudha yudhanto, a. w. (2017). *mudah membuat dan berbisnis aplikasi android dengan android studio*. jakarta: Pt Elex media komputindo.

Zuhdi, M., 1993, *Mashail Fiqhiyah*, Haji masagung, Jakarta.

Zufria, I. (2016). Pemodelan Berbasis UML (Unified Modeling Language) dengan Strategi Teknik Orientasi Objek User Centered Design (UCD) dalam Sistem Administrasi Pendidikan. *Research Gate*, (August). Retrieved from <https://scholar.google.co.id/citations?user=vfErXkMAAAAJ&hl=id>

SOURCE CODE

MainActivity

```
package com.example.kalkulatorzakat;

import android.content.DialogInterface;
import android.content.Intent;
import android.os.Bundle;
import
com.google.android.material.floatingactionbutton.FloatingActionButton;
import
com.google.android.material.snackbar.Snackbar;
import android.view.View;

import
androidx.appcompat.app.AlertDialog;
import androidx.core.view.GravityCompat;
import
androidx.appcompat.app.ActionBarDrawerToggle;
import android.view.MenuItem;
import
com.google.android.material.navigation.NavigationView;
import
androidx.drawerlayout.widget.DrawerLayout;

import
androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import android.view.Menu;
import android.widget.ImageButton;

public class MainActivity extends
AppCompatActivity
    implements
NavigationView.OnNavigationItemSelectedListener {

    ImageButton btn_1, btn_2, btn_3,
    btnkeluar,
    btninternak, btnnemas, btnprofesi, btnfitrah;

    @Override
    protected void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);
        Toolbar toolbar =
findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);

        getSupportActionBar().setHomeButtonEnabled(true);
        FloatingActionButton fab =
fab.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Snackbar.make(view, "Replace
with your own action",
Snackbar.LENGTH_LONG)
                    .setAction("Action",
null).show();
            }
        });
        DrawerLayout drawer =
findViewById(R.id.drawer_layout);
        NavigationView navigationView =
findViewById(R.id.nav_view);
        ActionBarDrawerToggle toggle = new
ActionBarDrawerToggle(
            this, drawer, toolbar,
R.string.navigation_drawer_open,
R.string.navigation_drawer_close);
        drawer.addDrawerListener(toggle);
        toggle.syncState();

        navigationView.setNavigationItemSelectedListener(this);

        //inisial timbol
        btn_1 =
(ImageButton)findViewById(R.id.btn_1);
        btn_2 =
(ImageButton)findViewById(R.id.btn_2);
        btn_3 =
(ImageButton)findViewById(R.id.btn_3);
        btnkeluar =
(ImageButton)findViewById(R.id.btnkeluar);
        btninternak =
(ImageButton)findViewById(R.id.btninternak);
        btnnemas =
(ImageButton)findViewById(R.id.btnnemas);
        btnprofesi =
```

```

(ImageButton)findViewById(R.id.btnprofesi
);
    btnfitrah =
(ImageButton)findViewById(R.id.btnfitrah);

    //fungsi tombol
    btn_1.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent iZakat = new
Intent(getApplicationContext(),zakat.class);
        startActivity(iZakat);
    }
});

    btn_2.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent ihitungfidyah = new
Intent(getApplicationContext(),hitungfidyah
.class);
        startActivity(ihitungfidyah);
    }
});
    btn_3.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent ifidyah = new
Intent(getApplicationContext(),fidyah.class)
;
        startActivity(ifidyah);
    }
});
    btninternak.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent izakatternak= new
Intent(getApplicationContext(),zakatternak.c
lass);
        startActivity(izakatternak);
    }
});
    btnemas.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent izakatemas= new
Intent(getApplicationContext(),zakatemas.cl
ass);
        startActivity(izakatemas);
    }
}

```

```

});
    btnfitrah.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent izakatfitrah= new
Intent(getApplicationContext(),zakatfitrah.cl
ass);
        startActivity(izakatfitrah);
    }
});
    btnprofesi.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent izakatprofesi= new
Intent(getApplicationContext(),zakatprofesi.
class);
        startActivity(izakatprofesi);
    }
});
    btnkeluar.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View v) {
        finish();
    }
});
}

@Override
public void onBackPressed() {
    DrawerLayout drawer =
findViewById(R.id.drawer_layout);
    if
(drawer.isDrawerOpen(GravityCompat.STAR
T)) {

drawer.closeDrawer(GravityCompat.STAR
T);
    } else {
        super.onBackPressed();
    }
}

@Override
public boolean
onOptionsItemSelected() {
    // Inflate the menu; this adds items to
the action bar if it is present.
    getMenuInflater().inflate(R.menu.main,
menu);
    return true;
}
}

```

```

@Override
public boolean
onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here.
    The action bar will
    // automatically handle clicks on the
    Home/Up button, so long
    // as you specify a parent activity in
    AndroidManifest.xml.
    int id = item.getItemId();

    //noinspection SimplifiableIfStatement
    if (id == R.id.action_settings) {
        showAbout();
    }

    return
    super.onOptionsItemSelected(item);
}

```

```

@SuppressWarnings("StatementWithEmpty
Body")
@Override
public boolean
onNavigationItemSelected(MenuItem item)
{
    // Handle navigation view item clicks
    here.
    int id = item.getItemId();

    if (id == R.id.dalil) {
        Intent imenuzakat = new
        Intent(getApplicationContext(),menuzakat.cl
        ass);
        startActivity(imenuzakat);
    } else if (id == R.id.tentangzakat) {
        Intent imenuzakat3 = new
        Intent(getApplicationContext(),menuzakat3.
        class);
        startActivity(imenuzakat3);
    } else if (id == R.id.hartazakat) {
        Intent imenuzakat2 = new
        Intent(getApplicationContext(),menuzakat2.
        class);
        startActivity(imenuzakat2);
    } else if (id == R.id.penerimazakat) {
        Intent imenuzakat1 = new
        Intent(getApplicationContext(),menuzakat1.
        class);
        startActivity(imenuzakat1);
    } else if (id == R.id.dalilfidyah) {
        Intent iactivity1 = new
        Intent(getApplicationContext(),Activity1.cla
        ss);

```

```

        startActivity(iactivity1);
    } else if (id == R.id.penerimafidyah) {
        Intent iactivity2 = new
        Intent(getApplicationContext(),Activity2.cla
        ss);
        startActivity(iactivity2);
    } else if (id == R.id.nav_share) {

    } else if (id == R.id.nav_send) {
        showAbout();
    }

    DrawerLayout drawer =
    findViewById(R.id.drawer_layout);

    drawer.closeDrawer(GravityCompat.STAR
    T);
    return true;
}

```

```

void showAbout() {
    new
    AlertDialog.Builder(MainActivity.this)
        .setTitle("Tentang")
        .setMessage("Aina Amelia Zein
        (1997-09-26)\n" +
        "niavarhen32@gmail.com\n"
        +
        "Universitas Islam Negeri
        Sumatera Utara")
        .setPositiveButton("Ok", new
        DialogInterface.OnClickListener() {
            @Override
            public void
            onClick(DialogInterface dialog, int which) {

            }
        })
        .show();
}

<?xml version="1.0" encoding="utf-8"?>
<androidx.drawerlayout.widget.DrawerLayo
ut
xmlns:android="http://schemas.android.com
/apk/res/android"

xmlns:app="http://schemas.android.com/apk
/res-auto"

xmlns:tools="http://schemas.android.com/to
ols"
    android:id="@+id/drawer_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"

```

```

        android:fitsSystemWindows="true"
        tools:openDrawer="start">

        <include
            layout="@layout/app_bar_main"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
        />

        <com.google.android.material.navigation.NavigationView
            android:id="@+id/nav_view"
            android:layout_width="wrap_content"
            android:layout_height="match_parent"
            android:layout_gravity="start"
            android:fitsSystemWindows="true"

            app:headerLayout="@layout/nav_header_main"

            app:menu="@menu/activity_main_drawer"
        />

    </androidx.drawerlayout.widget.DrawerLayout>

```

HitungFidyah

```

package com.example.kalkulatorzakat;

import android.content.DialogInterface;
import android.os.Bundle;
import android.text.Editable;
import android.text.TextUtils;
import android.text.TextWatcher;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;

import java.text.DecimalFormat;
import java.text.NumberFormat;
import java.util.Locale;

public class hitungfidyah extends AppCompatActivity {
    private EditText editPuasa, editBeras;

```

```

        private Button btnHitung;
        private TextView txtLuas;

        @Override
        protected void onCreate(Bundle savedInstanceState) {
            super.onCreate(savedInstanceState);

            setContentView(R.layout.activity_hitungfidyah);

            getSupportActionBar().setHomeButtonEnabled(true);

            // kenalkan komponen-komponen yang ada di layout activity_main
            editPuasa = (EditText)
                findViewById(R.id.editpuasa);
            editBeras = (EditText)
                findViewById(R.id.editberas);
            btnHitung = (Button)
                findViewById(R.id.btn_htg);
            txtLuas = (TextView)
                findViewById(R.id.txt_luas);

            getSupportActionBar().setTitle("Hitung Fidyah");

            editPuasa.addTextChangedListener(new TextWatcher() {

                private String current = "";

                @Override
                public void
                    beforeTextChanged(CharSequence s, int start, int count, int after) {

                }

                @Override
                public void
                    onTextChanged(CharSequence s, int start, int before, int count) {

                }

                @Override
                public void
                    afterTextChanged(Editable s) {

                    if (!s.toString().equals(current)) {

                        editPuasa.removeTextChangedListener(this)

```

```

;

        String replaceable =
String.format("[%s,\\s]", "Rp");
        String cleanString =
s.toString().replaceAll(replaceable, "");

        double parsed;
        try {
            parsed =
Double.parseDouble(cleanString);
        } catch
(NumberFormatException e) {
            parsed = 0.00;
        }
        NumberFormat formatter =
NumberFormat.getInstance(new
Locale("id", "ID"));

formatter.setMaximumFractionDigits(0);
        String formatted =
formatter.format((parsed));

        current = formatted;
        editPuasa.setText(formatted);

editPuasa.setSelection(formatted.length());

editPuasa.addTextChangedListener(this);

    }
}
});

editBeras.addTextChangedListener(new
TextWatcher() {

    private String current = "";

    @Override
    public void
beforeTextChanged(CharSequence s, int
start, int count, int after) {

    }

    @Override
    public void
onTextChanged(CharSequence s, int start,
int before, int count) {

    }

    @Override

```

```

        public void
afterTextChanged(Editable s) {

            if (!s.toString().equals(current)) {

editBeras.removeTextChangedListener(this)
;

                String replaceable =
String.format("[%s,\\s]", "Rp");
                String cleanString =
s.toString().replaceAll(replaceable, "");

                double parsed;
                try {
                    parsed =
Double.parseDouble(cleanString);
                } catch
(NumberFormatException e) {
                    parsed = 0.00;
                }
                NumberFormat formatter =
NumberFormat.getInstance(new
Locale("id", "ID"));

formatter.setMaximumFractionDigits(0);
                String formatted =
formatter.format((parsed));

                current = formatted;
                editBeras.setText(formatted);

editBeras.setSelection(formatted.length());

editBeras.addTextChangedListener(this);

            }
        }
});

        // berikan action button hitung untuk
menghitung hasil
        btnHitung.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String puasa, beras;
                puasa =
removeComma(editPuasa.getText().toString
());
                beras =
removeComma(editBeras.getText().toString
());

                // TextUtils berfungsi supaya
ketika EditText nya tidak terisi (Kosong)

```

```

Maka
    // nanti akan muncul notifikasi
    "tidak boleh kosong"
    if (TextUtils.isEmpty(puasa)) {
        editPuasa.setError("Tidak Boleh
Kosong!!!");
        editPuasa.requestFocus();
    } else if
(TextUtils.isEmpty(beras)) {
        editBeras.setError("Tidak Boleh
Kosong!!!");
        editBeras.requestFocus();
    } else {
        // Masukan Rumus untuk
menghitung Panjang dan juga lebar nya
        double p =
Double.parseDouble(puasa);
        double l =
Double.parseDouble(beras);
        double hasil = p * l;
        Log.d("CAK", "hasil: "+hasil+",
puasa: "+puasa+", beras : "+beras);
        // Kemudian Hasil di tampilkan
di TextView
        //          txtLuas.setText("Jumlah
Fidyah yang Harus Dibayar : Rp. " + hasil);

        DecimalFormat formatter = new
DecimalFormat("#,###");
        String formatted =
formatter.format(hasil);
        showDialog("Jumlah Fidyah
yang Harus Dibayar", "Rp. " +formatted);
    }
}
});
}

void showDialog(String title, String
message) {
    new
AlertDialog.Builder(hitungfidyah.this)
        .setTitle(title)
        .setMessage(message)
        .setPositiveButton("Ok", new
DialogInterface.OnClickListener() {
            @Override
            public void
onClick(DialogInterface dialog, int which) {

                }
            })
        .show();
}

```

```

String removeComma(String string) {
    int ch;
    StringBuilder string1 = new
StringBuilder();
    for (int i=0;i<string.length();i++) {
        ch = (int)string.charAt(i);
        if (ch<=57 && ch>=48) {
            string1.append(string.charAt(i));
        }
    }
    return string1.toString();
}
}

```

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com
/apk/res/android"

```

```

xmlns:app="http://schemas.android.com/apk
/res-auto"

```

```

xmlns:tools="http://schemas.android.com/to
ols"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"

```

```

    android:background="@android:color/white
"

```

```

    app:layout_behavior="@string/appbar_scrol
ling_view_behavior"
    tools:context=".hitungfidyah">

```

```

<ImageView
    android:id="@+id/imageView3"
    android:layout_width="match_parent"
    android:layout_height="150dp"
    android:background="@drawable/dw2"
/>

```

```

<LinearLayout
    android:orientation="vertical"
    android:layout_width="match_parent"

    android:layout_height="match_parent">

```

```

        <TextView

        android:layout_width="wrap_content"

```

android:layout_height="wrap_content"

android:layout_marginLeft="15dp"

android:layout_marginTop="25dp"
 android:text="APLIKASI"
 android:textSize="10sp" />

<TextView

style="@android:style/Widget.TextView"
 android:layout_width="171dp"

android:layout_height="wrap_content"

android:layout_marginLeft="40dp"
 android:layout_marginTop="0dp"
 android:text="Hitung"

android:textColor="@android:color/black"
 android:textSize="35dp"
 android:textStyle="bold" />

<TextView

android:layout_width="wrap_content"

android:layout_height="wrap_content"

android:layout_marginLeft="80dp"
 android:layout_marginTop="0dp"
 android:text="Fidyah"

android:textColor="@android:color/black"
 android:textSize="35dp"
 android:textStyle="bold" />

</LinearLayout>

<LinearLayout

android:layout_width="match_parent"

android:layout_height="wrap_content"
 android:layout_marginTop="200dp"
 android:layout_marginLeft="10dp"
 android:layout_marginRight="10dp"
 android:layout_weight="3"
 android:orientation="vertical">

<TextView

 android:text="Biaya 1 (satu) kali makan"

 android:textStyle="bold"
 android:textSize="15dp"

android:layout_width="wrap_content"

android:layout_height="wrap_content"/>

<EditText

 android:id="@+id/editpuasa"

android:layout_width="match_parent"

android:layout_height="wrap_content"
 android:layout_weight="1"
 android:maxLength="10"
 android:ems="10"

android:inputType="numberDecimal"
/>

<TextView

 android:text="Jumlah Puasa Yang Ditingalkan"

android:layout_marginTop="10dp"
 android:textStyle="bold"
 android:textSize="15dp"

android:layout_width="wrap_content"

android:layout_height="wrap_content"/>

<EditText

 android:id="@+id/editberas"

android:layout_width="match_parent"

android:layout_height="wrap_content"
 android:layout_weight="1"
 android:ems="10"
 android:maxLength="5"

android:inputType="numberDecimal"
/>

<Button

 android:id="@+id/btn_htg"
 android:layout_width="200dp"

android:background="@android:color/holo_green_dark"

android:textColor="@android:color/white"

```

        android:layout_height="wrap_content"
        android:layout_gravity="center"

```

```

        android:layout_marginTop="40dp"
        android:text="hitung"/>

```

```

        <TextView

```

```

        android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:id="@+id/txt_luas"
        android:text=""

```

```

        android:layout_marginTop="15dp"

```

```

        android:layout_marginBottom="8dp"
        android:textSize="25dp"
        android:textStyle="bold"/>

```

```

    </LinearLayout>

```

```

</RelativeLayout>

```

Hitung Zakat

```

package com.example.kalkulatorzakat;

```

```

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.ImageButton;
import
androidx.appcompat.app.AppCompatActivity;

```

```

public class hitungzakat extends
AppCompatActivity {
    private ImageButton
btninternak,btnemas,btnprofesi,btnfitrah;

```

```

    @Override
    protected void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);

```

```

    setContentView(R.layout.activity_hitungzak
at);

```

```

    getSupportActionBar().setHomeButtonEnab
led(true);

```

```

        btninternak =
(ImageButton)findViewById(R.id.btninternak

```

```

);
        btnemas =
(ImageButton)findViewById(R.id.btnemas);
        btnprofesi =
(ImageButton)findViewById(R.id.btnprofesi
);
        btnfitrah =
(ImageButton)findViewById(R.id.btnfitrah)
;

```

```

        btninternak.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent izakatinternak= new
Intent(getApplicationContext(),zakatinternak.c
lass);

```

```

                startActivity(izakatinternak);
            }
        });
        btnemas.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent izakatemas= new
Intent(getApplicationContext(),zakatemas.cl
ass);

```

```

                startActivity(izakatemas);
            }
        });
        btnfitrah.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent izakatfitrah= new
Intent(getApplicationContext(),zakatfitrah.cl
ass);

```

```

                startActivity(izakatfitrah);
            }
        });
        btnprofesi.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent izakatprofesi= new
Intent(getApplicationContext(),zakatprofesi.
class);

```

```

                startActivity(izakatprofesi);
            }
        });

```

```

    }
}
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout

```



```
xmlns:android="http://schemas.android.com/apk/res/android"
```

```
xmlns:app="http://schemas.android.com/apk/res-auto"
```

```
xmlns:tools="http://schemas.android.com/tools"
```

```
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="#F0930A"
```

```
app:layout_behavior="@string/appbar_scrolling_view_behavior"
    tools:context=".hitungzakat"
>
```

```
<ImageView
    android:id="@+id/imageView3"
    android:layout_width="match_parent"
    android:layout_height="200dp"
    android:background="@drawable/dwl"
/>
```

```
<ImageView
    android:layout_width="300dp"
    android:layout_height="160dp"
    android:layout_marginLeft="200dp"
    android:layout_marginTop="40dp"
```

```
android:background="@drawable/tangan"
/>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
```

```
<LinearLayout
```

```
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1"
    android:orientation="vertical">
```

```
<TextView
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_marginLeft="15dp"
```

```
    android:layout_marginTop="25dp"
    android:text="APLIKASI"
    android:textSize="10sp" />
```

```
<TextView
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_marginLeft="80dp"
```

```
    android:layout_marginTop="50dp"
    android:text="Hitung"
```

```
    android:textColor="@android:color/black"
    android:textSize="35dp"
    android:textStyle="bold" />
```

```
</LinearLayout>
```

```
<LinearLayout
```

```
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:layout_marginLeft="50dp"
    android:layout_marginTop="30dp"
    android:layout_marginRight="50dp"
    android:layout_weight="2"
    android:orientation="horizontal">
```

```
<LinearLayout
```

```
    android:layout_width="0dp"
    android:layout_height="280dp"
```

```
    android:layout_marginRight="0dp"
    android:layout_weight="1"
```

```
    android:background="@android:color/white"
```

```
    android:gravity="center_horizontal"
    android:orientation="vertical">
```

```
<ImageButton
```

```
    android:id="@+id/btinternak"
    android:layout_width="80dp"
    android:layout_height="80dp"
```

```
    android:layout_marginTop="35dp"
    android:background="#000"
```

```

android:src="@drawable/emas2" />

<TextView

style="@android:style/Widget.TextView"

android:layout_width="wrap_content"

android:layout_height="wrap_content"
    android:gravity="center"
    android:text="Zakat Ternak"

android:textColor="@android:color/black"
    android:textSize="10dp"
    android:textStyle="bold" />

<ImageButton
    android:id="@+id/btnemas"
    android:layout_width="80dp"
    android:layout_height="80dp"

android:layout_marginTop="10dp"
    android:background="#000"
    android:src="@drawable/ungu"
/>

<TextView

style="@android:style/Widget.TextView"

android:layout_width="wrap_content"

android:layout_height="wrap_content"
    android:gravity="center"
    android:text="Zakat Emas"

android:textColor="@android:color/black"
    android:textSize="10dp"
    android:textStyle="bold" />

</LinearLayout>

<LinearLayout
    android:layout_width="0dp"
    android:layout_height="280dp"
    android:layout_marginLeft="0dp"
    android:layout_weight="1"

android:background="@android:color/white"

android:gravity="center_horizontal"
    android:orientation="vertical">

```

```

<ImageButton
    android:id="@+id/btnfitriah"
    android:layout_width="80dp"
    android:layout_height="80dp"

android:layout_marginTop="35dp"
    android:background="#000"

android:src="@drawable/fidyah" />

<TextView

style="@android:style/Widget.TextView"

android:layout_width="wrap_content"

android:layout_height="wrap_content"
    android:gravity="center"
    android:text="Zakat Fitrah"

android:textColor="@android:color/black"
    android:textSize="10dp"
    android:textStyle="bold" />

<ImageButton
    android:id="@+id/btnprofesi"
    android:layout_width="80dp"
    android:layout_height="80dp"

android:layout_marginTop="10dp"
    android:background="#000"

android:src="@drawable/kalkulator2" />

<TextView

style="@android:style/Widget.TextView"

android:layout_width="wrap_content"

android:layout_height="wrap_content"
    android:gravity="center"
    android:text="Zakat Profesi"

android:textColor="@android:color/black"
    android:textSize="10dp"
    android:textStyle="bold" />

</LinearLayout>

</LinearLayout>
</LinearLayout>
</RelativeLayout>

```

